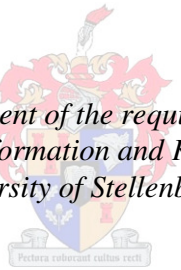
The role of knowledge management in offshore outsourced software development

by
Andile A Swartbooi



Thesis presented in fulfilment of the requirements for the degree of Master of Philosophy (Information and Knowledge Management) University of Stellenbosch

Supervisor: Mr Daniel F Botha
Faculty of Arts and Social Sciences
Department of Information Science

December 2010

Declaration

By submitting this thesis/dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

December 2010

Abstract

In an effort to streamline operations and focus on what they regard as core activities, a growing number of organizations from both developed and developing countries are increasingly looking to outsource their software development and maintenance activities to lower cost countries such as India and China, this is evidenced by the phenomenal growth in India's software industry and the number of major overseas IT companies establishing subsidiaries and relocating their Research and Development operations to India's high-tech cities such as Hyderabad, Chennai and Pune. With the mere size of their populations standing at over a billion people each, supported by their governments, Indian and Chinese business have been able to leverage this population advantage producing a large pool of software engineers, technical specialists and back office workers to cater for the talent demands of the world.

While the actual software development process might be non-core to many organizations, it however yields software applications that drive critical business processes and embed valuable organizational knowledge. The handing over of software development operations by an organization to a third party poses a risk of creating a dependency and exposing vital business knowledge to competition thereby compromising its competitive edge. Both the people that participate in software development projects and the software products these people develop possess knowledge which need to be secured and leveraged to enable the continued success of an organization. Securing these knowledge artefacts and the knowledge created by the software development lifecycle process cannot be left to chance, therefore the success of an organization's software development activities needs to be measured largely on its ability to secure knowledge assets that derive from such process and the leveraging of such knowledge to drive organizational strategy and yield new knowledge.

This thesis is premised on the fact that knowledge is the one competitive advantage that separates successful nations from failed states and one dominant force that prevails across all successful economies in the 21st century, hence the notion of a knowledge economy.

The study seeks to understand the importance of the role played by knowledge in an outsourced software development engagement and how knowledge management affects the success of this engagement. By exploring the business drivers that spur organizations to outsource their IT activities, the software development lifecycle, the different outsource models available to organizations and the inherent risks surrounding knowledge loss, the thesis seeks to gain an understanding and the criticality of managing knowledge within an outsourced software development context and the strategies that organizations can utilize to deliver on outsourcing promises with minimal risk.

Opsomming

Ten einde hulle werksaamhede meer vaartbelyn te maak en om ingestel te bly op dit wat hulle as kernbedrywighede beskou, kyk al hoe meer organisasies in ontwikkelde en ontwikkelende lande na die moontlikheid om die ontwikkeling en instandhouding van hulle sagteware uit te kontrakteer na lande soos Indië en China, waar dit goedkoper gedoen kan word as tuis. Dié feit blyk duidelik uit die fenomenale groei in veral Indië se sagtewarenywerheid en die getal groot oorsese IT firmas wat hulle navorsing en ontwikkeling in hoë-tegnologie stede soos Hyderabad, Chennai en Puna laat doen. Met bevolkings van meer as ’n miljard elk, kon Chinese en Indiese ondernemings hierdie voorsprong benut om ‘n magdom sagteware-ingenieurs, tegniese spesialiste en kantoorwerkers te produseer om in die wêreld se vraag na kundigheid te voorsien.

Terwyl die ontwikkeling van sagteware miskien nie deur baie ondernemings as ‘n kernbedrywigheid beskou word nie, lewer dit tog aanwendings op wat kritieke sakeaktiwiteite aandryf en waardevolle organisatoriese kennis vasvang. Die oordra van sagteware-ontwikkeling van een onderneming na ‘n derde party gaan egter gepaard met die risiko dat dit afhanklikheid kan skep en ook uiters belangrike sakekennis aan konkurrente toeganklik maak, wat die mededingende voorsprong wat sulke kennis bied bedreig. Die mense betrokke by die ontwikkeling van sagteware en die produkte wat hulle sodoende skep, is ‘n bron van kennis wat beveilig en verveelvuldig moet word om ‘n onderneming in staat te stel om suksesvol te bly voortbestaan. Die versekering van hierdie verworwe kennis en die kundigheid wat deur die ontwikkelingsiklus van die sagteware geskep word, kan nie aan die toeval oorgelaat word nie – die sukses van ‘n onderneming se sagteware-ontwikkeling moet veral gemeet word aan sy vermoë om die kennisbates wat uit die proses voortvloei te verseker, en om hierdie kennis te verveelvuldig om organisatoriese strategieë aan te dryf en nuwe kennis op te lewer.

Hierdie tesis se uitgangspunt is dat kennis die mededingende voorsprong is wat suksesvolle nasies onderskei van die res; dit is dié faktor wat kenmerkend is van al die suksesvolle ekonomieë van die 20ste eeu, en die kern van die begrip van ‘n “kennis-ekonomie”.

Hierdie ondersoek wil die belangrikheid verken van die rol wat gespeel word deur kennis in ‘n uitgekontrakteerde verbintenis vir die ontwikkeling van sagteware and hoe kennisbestuur die

sukses van so ‘n verbintenis affekteer. Deur ondersoek in te stel na die motivering wat besighede aanspoor om hulle IT bedrywighede uit te plaas, na die sagteware-ontwikkeling lewenssiklus, die verskillende modelle van uitkontraktering wat vir organisasies beskikbaar is en die inherente risiko’s rondom kennisverlies, wil hierdie tesis ‘n begrip vorm van die kritieke noodsaaklikheid vir die bestuur van kennis in ‘n uitgekontrakteerde sagteware-ontwikkeling en van die strategieë wat organisasies kan aanwend om die voordele wat uitkontraktering beloof ten volle te benut teen minimale risiko.

Acknowledgements

My sincere gratitude goes to my supervisor D.F Botha for his invaluable advice and support, his motivation and encouragement during the review sessions kept me going.

I would like to thank Prof. Johan Kinghorn for the encouraging emails to the group.

This thesis is dedicated to my family who stood by me sacrificing weekends, lots of holidays and family time while I was working on this project. Special thanks goes to my daughter Sive, who kept me going by constantly asking me, “How’s your book going, Dad?”

List of Abbreviations

BCP	Business Continuity Planning.
BOT	Build Operate and Transfer.
BRD	Business Requirements Document.
CBT	Computer Based Training.
CMS	Content Management System.
CIO	Chief Information Officer.
CISSP	Certified Information Systems Security Professional.
CKO	Chief Knowledge Officer.
CMM	Capability Maturity Model.
CMMi	Capability Maturity Model Integration.
COBOL	Common Business Oriented Language.
COBIT	Control Objectives for Information and related Technology.
COP	Communities of Practice.
CRM	Customer Relationship Management.
CSA	Control Self Assessment.
DMS	Document Management System.
DRP	Disaster Recovery Planning.
ERP	Enterprise Resource Planning.
HR	Human Resources.
IBM	International Business Machines.
ICT	Information and Communication Technology.
IDE	Integrated Development Environment.
ISACA	Information Systems Audit and Control Association
(ISC) ²	International Information Systems Security Certification Consortium.
IT	Information Technology.
ITSEC	Information Technology Security Evaluation Criteria.
J2EE	Java2 Enterprise Edition.
KM	Knowledge Management.
KMS	Knowledge Management Systems.

KPMG	Klynveld Peat Marwick & Goerdeler.
LAN	Local Area Network.
LMS	Learning Management System.
MAN	Metropolitan Area Network.
OSI	Open System Interconnection.
PDA	Personal Digital Assistant.
PIM	Personal Information Management.
PKI	Public Key Infrastructure.
PMBOK	Project Management Body of Knowledge.
RDBMS	Relational Database Management System.
SDLC	Software Development Lifecycle.
SECI	Socialization Externalization Combination Internalization.
SEI	Software Engineering Institute.
SOX	Sarbanes Oxley.
SPCC	Software Project Control Centre.
SRS	Software Requirements Specifications.
TPS	Transaction Processing System.
TCSEC	Trusted Computer Security Evaluation Criteria.
VPN	Virtual Private Network.
WWW	World Wide Word.
WAN	Wide Area Network.

List of Figures

List of Tables

Contents

Chapter 1
Introduction

1.1 Introduction

As organizations evolve and mature in the process of delivering value to their customers, they are faced with a challenge of competing with smarter competitors under tough economic conditions utilizing ever changing technological solutions. The breaking down of economic borders as a result of globalization brings to organizations advantages of entering new markets while posing on the other hand challenges of competing with more advanced economies operating under less regulated conditions. These opposing pressures have forced organizations to review their business models and to look at ways in which they can fast-track the growth of their delivery capabilities in order to satisfy growing demand while curbing costs so as to remain competitive. While the traditional way of growing organizational capabilities has been to invest in internal capacity and to implement ways of retaining these capabilities through such strategies as human resource retention and intellectual property protection, globalization has opened up an avenue for organizations to buy readymade capability and to leverage this to deliver on their growth and innovation strategies. Outsourcing is therefore one of the key strategies that organizations can use to rapidly gain access to critical resources and know-how needed to implement strategies and deliver service. Unfortunately buying critical capability such as know-how is not a simple over the counter transaction, it is a strategic decision laden with risks which if not managed and mitigated effectively could threaten the viability of an organization.

This chapter introduces the concepts of outsourcing and knowledge management and establishes a common understanding of these two topics as they pertain to offshore software development. These concepts are explained with a view to providing background and context to this thesis. Taking into consideration the fact that outsourcing does not take place in isolation, a highlevel examination of drivers that determine and influence outsourcing decisions is made, also, models of outsourcing and associated risks are reviewed.

1.2 Research background

In an attempt to deal with rising IT costs and a limited pool of internal IT skills that is stretched between business's increasing demand for new technology solutions and the support of existing systems, more and more companies are looking to places such as India and China as a source of skilled, relatively cheap pool of IT resources. While in the beginning location was paramount and geographic proximity of the vendor to the outsourcing client was a key component in the outsourcing decision, the removal of boundaries due to advances in Information and Communication Technology (ICT) has made outsourcing to these offshore countries a viable option[1], companies are now able to access skills in virtually every part of the world, sometimes at a fraction of local cost.

Financial gain in the outsourcing model is higher when the work is performed offshore, the record year on year profitability of leading Indian outsource companies[2] is evidence of this growing trend to outsource work offshore.

The aging IT workforce, the constant change in software development technologies and business's increasing reliance on information technology for its operations, products and services exacerbates the need for organizations to look externally for talent. With increasing competition putting pressure on their bottom line, organizations often have to compete on price to stay afloat as a result offshore outsourcing promises to reduce the organization's highest overhead, its workforce costs. In a drive to become lean by focusing on core business, organizations are looking to outsourcing to rid themselves of noncore activities. While IT services are essential to the success of organizations, for many, IT activities and more specifically software development fall within the noncore category. When IT activities are outsourced to a third party, the outsourcing organization gains immediate access to specialist skills that can perform IT tasks more efficiently, this frees up internal resources to focus on refining and perfecting core activities.

From the above observations it can be seen that IT outsourcing has matured to become a business strategy rather than a once off stop-gap arrangement and organizations who manage this process well can gain advantage over competition.

---

[1] Friedman T, 2005

[2] Gold T, 2005

While promising to deliver on the bottom line profits, outsourcing brings with it risks with respect to merging of client/vendor corporate cultures, exposure of the company's intellectual property to external parties and most importantly, potential loss of organizational knowledge which could compromise the organization's competitive urge and create overreliance on the vendor.

1.3 Knowledge management and outsourcing defined

1.3.1 What is knowledge?

Knowledge is information that enables actions and decisions[3]. Knowledge is distinguishable from Data (raw facts) and Information (contextualized data) in that it is a higher order of the two and is what informs the decisions and the actions of the one who possesses it, one will act in a particular way depending on the knowledge one possesses about a situation. While knowledge builds from available data and information, it is fundamentally different from data and information[4]. Knowledge enables one to sift through data and produce valuable information relevant to the task at hand, it also allows one to synthesize available information to produce more refined and relevant information on which decisions can be taken or a course of action can be followed.

Knowledge can be classified as either Explicit or Tacit[5]. Explicit knowledge refers to knowledge that is articulated in words and numbers through data, reports, books, manuals, audiovisual materials etc. On the other hand Tacit knowledge represent the difficult to express forms of knowledge such as insights, opinions, intuition etc. that is built mainly on personal experiences, observations and interactions.

1.3.2 Knowledge management

Organizations manage knowledge because they see it as a means of achieving their mission, they see knowledge as being key to their ability to compete and grow[6].

---

[3] Becerra-Fernandez I et al, 2004

[4] Wiig K, 1999

[5] Nonaka I & Takeuchi H, 1995

[6] O'Dell C & Elliot S, 2003

Knowledge management involves the management of the four key supporting processes, namely:-

- Knowledge discovery.
- Knowledge capture.
- Knowledge sharing.
- Knowledge application.

In the software development context the management of knowledge involves ensuring that knowledge created in the software development lifecycle is identified, captured and made available as learnings to a wider audience within the organization. Experiences from previous efforts also need to be made available to the development team for application in order to fast-track the process and to ensure that they do not reinvent the wheel.

Knowledge management will be revisited later on in the study and the Knowledge Management processes will be discussed in greater detail.

1.3.3 IT Outsourcing defined

IT Outsourcing is generally defined as selectively handing over to a vendor some or all of the IT functions, ranging from simple data entry to software development and maintenance, data centre operations and full systems integration[7], essentially outsourcing involves the transfer of responsibility by a client to a vendor to perform a particular task or service for a given period of time. Depending on the service criticality and response time requirements of the client, the outsource vendor can be located from as near as within the client premises to as far afield as offshore where offshore is defined as a location in another region of the world (not necessarily overseas). India and China are the respective leaders in the provisioning of offshore outsourced IT services[8].

[7] Kehal HS & Singh VP, 2006

[8] Accenture , 2008

1.4 Evolution of the IT outsourcing industry

1.4.1 The emergence of IT outsourcing

Ever since the end of the cold war and the subsequent fall of the Berlin Wall, globalization, fuelled by the drop in the cost of technology and improvements in communication technology has laid the foundation for cross-continent collaboration[9]. The proliferation of low cost high bandwidth telecommunications connectivity between India and the United States set the scene for a viable outsource industry that has matured over time to the levels experienced today. The outsourcing industry has grown wider than just pure IT activities to encompass such business functions as Call Centre operations, Back Office administration and hi-tech Research and Development. The advent of mega fast communication channels, integrated collaboration systems and sophisticated workflow software has meant that challenges posed by distance and time differences have largely been overcome as client and vendor are able to collaborate via telephony, email, video conferencing, instant messaging and such like as and when required. Globalization fuelled by the need to find new markets has meant that organizations have used outsourcing as a stepping stone towards establishing a footprint in the countries they are outsourcing to.

Predictions from industry analysts such as Gartner are that the wave of outsourcing will continue to grow and the structure and format of relationships will continue to evolve and mature.

1.4.2 Examining the drivers for software development outsourcing

While offshore IT outsourcing started as a cost cutting measure spurred mainly by the abundance of cheap specialist skills in India, it has since evolved into a strategic sourcing option that is considered and implemented by many organizations of varying sizes.
With hardware prices falling year on year and the computing power increasing exponentially, software remains the one area where costs have been steadily rising and as such it is not surprising that many organizations are targeting application software development for outsourcing, partly as a way of reducing acquisition and operating costs.

[9] Friedman T, 2005

IT outsourcing amongst many organizations is driven largely by Cost, Specials skills, Business demand, and the need for Agility[10]. Expanding further on this:-

- **Cost reduction.**
  Organizations realize that while cost might not be the only driver, it is however important that the outsource service is provided at a justifiable cost.
- **Access to specialist skills.**
  Organizations are finding it difficult to justify the investment required to keep up with ever changing technology and skills requirements, they are having to look to leveraging off vendor IT investment with regard to accessing latest technical skills.
- **Focusing on core competencies.**
  Organizations that have made a decision on the non-strategic status of IT would outsource their IT to a third party in order to focus more on the activities they regard as being core to their business.
- **Meeting business demand.**
  Organizations realize that they cannot always meet demand picks with internal resources and as such have to rely on external partners for additional capacity.
- **Enabling organizational agility.**
  In order to respond swiftly to market demands, organizations need varying capacity and skill sets which cannot always be met from the internal pool of resources hence the need to form strategic alliances with external specialist vendors.

1.4.3 IT Outsourcing models

Looking at the organizational drivers for outsourcing as listed above, it is clear that outsource relationships can range from meeting a specific need in time to satisfying a strategic requirement over a longer term. Organizations therefore need to make decisions on the strategies that best meet these varying needs. Depending on the focus, when outsourcing, an organization can adopt either a Contract-Out strategy or a Preferred-Contractor strategy[11]. Contracting out implies a single transaction where a specific deliverable e.g. the development of an accounting system, is

[10] Kehal HS & Singh VP, 2006
[11] Wilcocks L & Lacity MC, 1998

outsourced to a qualifying vendor and the duration of the relationship is linked to the delivery period. Preferred-Contractor strategy refers to an alliance where a specific vendor is contracted based on some predetermined criteria to provide outsource services to the client over a longer negotiated term. The bottom line being that an organizations can choose between making a once off outsource transaction with the best vendor available or leverage off an existing and continuing relationship with a preferred partner. As shall be expanded later, it is worth noting that the two types of strategic choices (contract out vs. preferred contractor) have different consequences with regard to knowledge management within an outsourced software development arrangement.

Having made the decision on which parts of their IT to outsource, organizations are further faced with a choice of allocating the outsourced function to either a single vendor or breaking the function up and selectively allocating responsibilities to different vendors according to their areas of specialization. The emerging industry trend is to contract selectively rather than lump all functions with one vendor and to do this over shorter periods[12] (3 years).

1.4.4 Risks involved in offshore software development outsourcing

While IT outsourcing has been growing steadily, the successes have not been guaranteed. As much as 80% of the outsource deals are deemed unsuccessful for various reasons[13], coupling this with industry research that software development projects have a success rate of less than 50%[14], suddenly the risks of offshore outsourced software development become apparent. Industry reports also abound with articles regarding high Indian staff turnover, increase in staff costs and potential political instability[15], all adding to the need for organizations to view offshore outsourced software development initiatives as potentially risky as they could be terminated abruptly. Organizations therefore need to understand the impact this risk could have on their knowledge assets embedded within the outsourced software development function, we will expand later on the relevance of knowledge management in this regard.

---

[12] Wilcocks L & Lacity MC, 1998

[13] Gartner , 2004

[14] Standish Group, 2009

[15] Gartner , 2009

While outsourcing software development offshore has significant cost benefits especially to developed countries with strong currencies, there are negatives that need to be considered, some of which include[16] :-

- Added transaction costs as a result of the need to monitor and manage the outsource contract.
- Increased risk due to geopolitical events in vendor countries.
- Increased risk due to the vendor's risk appetite that might not be aligned with the client's business requirements.
- Loss of internal knowledge as a result of staff fears, disillusionment and accelerated attrition.
- Threat of a vendor pull out and the inability to re-establish the service internally.
- Loss of flexibility due to stringent contracting conditions.

Judging from the above list, it stands to reason that while it might bring benefits to business, the outsourcing process is potentially risky and therefore needs to be constantly monitored and controlled such that an organization is not overly dependent and hence dominated by the outsource vendor. In order to ensure mitigation of risks and maintain balance of power, organizations have a responsibility to be fully aware of the knowledge that gets created and flows through the outsource relationship. Risk awareness is hampered by the fact that much of outsource reporting centres around announcements of outsource mega-deals, little gets reported on what happens afterwards regarding failures and unfulfilled expectations.

1.5 Importance of this study

In an organization, knowledge resides within the people as individual and groups, the artefacts in the form of practices, technologies and knowledge repositories as well as the organizational entities. Developing software involves each of the knowledge locations in that:-

- The development team uses its knowledge of the organization requirements, the tools and the business processes to build an IT solution.
- The solution built constitutes a knowledge artefact as it encompasses all the knowledge from the teams involved in the development of the solution.

[16] Wilcocks L & Lacity MC, 1998

- Documentation is produced regarding the design of the solution, its relationship to other technology solutions within the organization and how it should be used and maintained.
- The developed solution facilitates the flow of information within organizational entities expediting actions and decision making.

The above is an indication of how pervasive the process of software development is in the organization's knowledge management lifecycle and as such how important it is for an organization to consider the impact that outsourcing will have on its knowledge management strategy. This research study will appeal to the operational teams involved in delivering offshore software projects as well as to company executives who strategize and take decisions that might involve outsourcing.

1.5.1 Knowledge is a vital organizational resource

It can be argued that the most vital resource of today's enterprise is its collective knowledge residing in the minds of its people, customers and vendors, organizations are increasingly being valued on their intellectual capital[17]. During outsourcing an organization hands over its most vital resource namely its knowledge assets residing in its Hardware and Software technologies, it's Processes and its People to an external party, the vendor, for utilization, enhancement and safekeeping. It is therefore critical that Knowledge Management becomes a key part of the outsourcing decision.

1.5.2 The changing face of offshore outsourcing

India's advantage as an offshoring location is fast eroding – its attractiveness takes a hit with each passing day. The high staff turnover, the strengthening of the Rupee against international currencies and communication and time zone issues are all contributing to questions on the sustainability of the offshoring model[18]. While delivering on the economic benefits, the long term sustainability of the outsourcing/offshoring model cannot be guaranteed, it is for this reason that organizations need to ensure that every outsource transaction is underpinned by sound knowledge management principles to ensure continuity in the event of an unfavourable outcome. Software development, while not core to many organizations is a knowledge generating process

[17] Becerra-Fernandez et al, 2004

[18] King R, 2008

and therefore it's outsource needs to be aligned to the organizations knowledge management processes.

1.6 Conclusion

Knowledge is key to achieving business goals - as such knowledge management should be a consideration in every strategic decision taken within an organization. Organizations are using outsourcing as a means of building and improving on their core competencies by moving noncore activities to external vendors who can perform these tasks better.

This chapter provided a definition of the key concepts that will be used in this research namely: - Knowledge, Knowledge Management, Software development Outsourcing and Offshoring. A background was given of outsourcing models, drivers for outsourcing and the risks that accompany outsourcing; reference was also made to the strategic drivers behind outsourcing and the changing picture of outsourcing influenced by globalization.

The next chapter looks at the research methodology applied, the considerations taken in conducting the research and the rationale behind the methodology.

Chapter 2
Research

2.1 Introduction

The previous chapter introduced the concept of outsourcing and its maturity as organizations evolve through the challenges of globalization. Knowledge management was also defined and its importance in the success of an organization highlighted. The previous chapter also gave a view of the purpose and significance of this research in positioning knowledge management within the offshore software development context. This chapter articulates the research problem the study is attempting to solve, the approach chosen to conduct the study and the methodology to be used in constructing the thesis.. The chapter also gives an account of the literature researched highlighting the findings of the different authors with regards to offshore outsourced software development.

2.2 Research context

Outsourcing is booming and is a big source of revenue in India accounting for 7% of that country's Gross Domestic Product, but this is a double edged sword as Mitra[19] asserts in her article that the model is in for some challenges in years to come. With the figures quoted above it is clear that information technology is increasingly playing a bigger part in the operations of organizations; in an effort to become more streamlined and competitive; organizations are turning to outsourcing as a means of effectively managing highly specialized activities such as software development. While outsourcing might solve the challenges associated with lack of skills and delivery capacity in software development, it on the other hand exposes the organizations to the following key risks:-

- Loss of control of key operations.
- Potential loss of intellectual property.
- Loss of know-how.

[19] Mitra S, 2008

- Greater dependency on third parties.
- Threat to information security and confidentiality.
- Quality problems and Hidden costs.

2.3 Research problem

Problem definition is the result of a reasoning process conducted by the researcher in order to translate the phenomena to be examined into a researchable research problem[20]. The research problem the study seeks to address is premised on the real world challenges surrounding knowledge management during outsourcing as highlighted above. The common thread binding the risks associated with offshore software development outsourcing is the potential loss of knowledge. The less effort organizations make in managing knowledge during software development outsourcing, the more real the above risks and the more at stake the viability of the outsourcing organization.

2.3.1 Hypothesis

The thesis hypothesizes that if software development is outsourced offshore then knowledge management will be a critical factor in the successful delivery of such software projects.

2.3.2 Research objective

Booth et al[21] asserts that we do research whenever we gather information to answer a question that solves a problem. This research seeks to establish the significance of knowledge management in the success of offshore outsourced software development and to find strategies that organizations can utilize to mitigate associated risks and safeguard their knowledge assets.

The research will attempt to answer two fundamental albeit interlinked questions, namely:-

- What role does knowledge management play in the success of offshore outsourced software development?
- What knowledge management strategies can organizations employ to ensure the success of their offshore outsourced software development initiatives?

[20] Jonker J & Pennink B, 2010

[21] Booth et al, 2008

2.4 Research approach

In structuring this research a choice had to be made with regard to the approach or design to be followed in answering the research question. Two research approaches were considered namely empirical and non-empirical approaches. The research problem requires that a review be done on existing data on outsourced software development projects and the role knowledge plays in such projects. The aim of the research would be to draw inferences from the analysis on the role that knowledge management plays in the success of outsourced software projects. Based on the conceptual nature of the research question, a non-empirical approach has been selected as the base for this study. The research is based on a focused literature review covering software development, knowledge management, IT outsourcing and project management. A targeted selection of published material covering books; journals and industry publications was referenced. Included were references from the author's personal experiences and observations emanating from involvement in a number of software development outsourcing initiatives within the South African financial services industry.

2.4.1 Background to the approach

The research approach gives a view of how the study is to be conducted. The theoretical questions posed by the research problem as articulated above points to a non-empirical approach as being an approach that would best answer the research question. Mouton[22] lists the following studies as forming part of the non-empirical approach:-

- **Conceptual analysis.**

  This is the analysis of the meaning of words or concepts through clarification and elaboration of the different dimensions of meaning.

- **Theory-building studies.**

  These studies are aimed at developing new models and theories to explain particular phenomena.

- **Philosophical analysis.**

  These studies are aimed at analysing arguments in favour of or against a particular position; they develop substantive points of view about such topics as ethics and logic.

[22] Mouton J, 2001

- **Literature reviews.**

  Literature reviews provide an overview of scholarship in a certain discipline through an analysis of trends and debates.

Having analysed the different non-empirical approaches; it is evident that the literature review approach would be the most appropriate approach to follow in answering the research question articulated earlier.

2.4.2 Relevance of the approach to the research problem

The research problem embodies a conceptual question about the relevance of knowledge management in the success of outsourced software development. To answer a conceptual question requires a review of available body of knowledge[23] in the problem domain. The body of knowledge to be reviewed in this study includes:-

- Definitions of outsourcing
- Different theories and models in the field of outsourcing
- Different strategies and approaches used to manage knowledge
- Existing data and findings produced by previous research in the fields of outsourcing and knowledge management
- Tools and Systems used in the development of software, integration of cross-border teams and management of knowledge.

A literature review approach which involves the review; analysis and interpretation of literature covering pertinent areas in offshore software development and knowledge management as indicated above lends itself as the most appropriate approach to answering the research questions posed in this study.

[23] Mouton J, 2001

2.4.3 Research approach strengths and weaknesses

A literature review approach has been selected as the basis for this research study. Literature review involves selecting a representative sample of sources that is read in order to gain an understanding of a specific problem domain.

- Literature review strengths

  Literature review is essential in helping the researcher to understand the concepts, the issues and the debates surrounding the area being researched. It provides a solid grounding upon which an informed analysis can be made. Literature review provides the background to the current discourse and allows one to anticipate trends and future direction.

- Literature review weaknesses

  Knowledge and insight gained in a literature review is highly dependent on the representativity of the sample used. A literature review can at best only summarise existing body of knowledge[24], it produces limited views and insights in cases where the domain being researched is not well documented. While literature reviews will produce theoretical insights; these often need to be validated through empirical means. Selective interpretation sometimes creeps in; in cases where the researcher is biased; resulting in skewed analysis or incorrect assumptions being made.

2.5 Nature of this research study

This is a theoretical research that explores the subject from a number of angles namely Technical, Management and Academic, it aims to integrate these different perspectives into a study that answers the research question posed earlier and allow for the following inferences and conclusions to be drawn on the subject.

- **Technical perspective.**

  Software development is a technical process of building software applications; these are used by business to automate processes.

  Tools (Development tools and Technology infrastructure) are used by the project to build the software applications.

- **Management.**

  Project management is a methodology used to coordinate the activities of building the

[24] Mouton J, 2001

software applications. This is a delivery process that governs and manages all the activities involved in building the software application. Knowledge Management tracks and accounts for the discovery, capture, sharing and application of all knowledge required by the project and in turn generated by the project.

- **Academic.**

  The study is based on research principles which are academically proven. The structuring of this research follows an academic process and all sources are based on credible references and inferences supported by evidence. Both the software development and the knowledge management concepts are also scholarly in nature and as such this research adds to the body of knowledge that is out there for studying.

2.6 Scope of the literature review

The theoretical context of this thesis requires that an in-depth literature research be done on the prevailing views on software development off-shoring and its impact on organizational knowledge. The challenge however with reviewing literature for a topical subject such as outsourcing and off shoring is that the discourse is populated by many differing views and versions of reality and because the subject is current, it is dynamic and is changing on a daily basis. Outsourcing is a multibillion dollar industry[25]; therefore it is not surprising that economics will play a part in influencing some of the writings that are available on the public domain and as such objectivity can sometimes be questionable.

While one needs to keep an open mind on the pervasive ideas, one also needs to apply cognitive filters in order to make sense of the views expressed by the different contributors and to ensure that inputs considered are relevant to the research study. The aim is to review the literature, make sense of the views expressed by the different writers and find meaning which will be the frame of reference for the following chapters. Finding meaning is the product of sense making[26].

[25] Gartner , 2008

[26] Weick K, 1995

2.6.1 Strategic role of knowledge in organizational success

Nonaka writes in the Harvard Business Review of 1991 that in an economy where the only certainty is uncertainty, one source of lasting competitive advantage is knowledge and its manipulation[27]. Prusak states that knowledge, which includes the things an organization knows, how it uses what it knows and how fast it can know something new, is the only thing that offers an organization a competitive edge[28]. Stewart takes this further when he says that knowledge and its management are more valuable and more powerful than natural resources, big factories or fat bankrolls of money[29].

Becerra-Fernandez et al argues that the most vital resource of today's enterprise is the collective knowledge residing in the minds of an organization's employees, customers and vendors. Learning how to manage organizational knowledge has many benefits[30]. The employees, customers and vendors are the ecosystem that define processes, create products and form markets; therefore the above statement can be read to indicate that knowledge is the key ingredient to an organization's very existence. Knowledge help create innovative products which provide market differentiation for an organization, this in turn translate in better revenues and bigger market share.

Max Boisot maintains that knowledge has become more valuable than its physical carrier and as such knowledge has come to be viewed as an asset in its own right[31]. Executives are custodians of an organization's assets and therefore if knowledge is a critical asset of an organization as Boisot puts it, it is incumbent upon the organization's executives to ensure that their strategies incorporate the management of this resource. Boisot introduces the concept of entropy where through knowledge organizations are able to "squeeze" more work out of a stock of free energy. These efficiencies brought about by knowledge management give the organization an edge over its competitors. With the move to a greener environment, organizations through knowledge can deliver more with less energy consumption; this becomes even more critical as green environment requirements become legislated. Boisot however cautions that competitive

[27] Nonaka I, 1991
[28] Prusak L, 1996
[29] Stewart T, 1999
[30] Becerra-Fernandez et al, 2004
[31] Boisot M, 1999

advantage does not flow automatically from possession of knowledge assets; an organization has to know how to extract value from these, in other words an organization's strategy plays a key role in leveraging the knowledge the organization possesses. The point coming across quite strongly in these writers' assertions and the consistency of their language is the importance of knowledge as a strategic component of any organization's success. Paying less attention to the management of its knowledge therefore can be equated to under-investing; which leads to an organization not reaching its full potential despite everyone working tremendously hard[32].

2.6.2 Locating knowledge within the software development process

Waterson[33] et al describe software development as involving a variety of cognitive and organizational issues concerning the communication and coordination of knowledge relating to the development project, the methodologies used, the domain area the software is targeted for and other organizational practices such as project logistics and reporting lines. Becerra-Ferndandez et al[34] assert that, apart from storing data and information, computer based information systems store knowledge as well. Laudon & Laudon[35] indicates that software enables data to be used by multiple functions for precise organizational coordination and control. Software automates work that would otherwise be performed manually and as such the knowledge needed to perform these tasks and the business rules needed to make the necessary decisions are captured into the software application. Mega applications like Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) and Supply Chain Management are such applications where business best practices (knowledge) are embodied into the software application. According to Florida[36] the principal factors for successful software development are *Talent, Knowledge* and *Intellectual Capital.* Talent comes from the individuals involved in the development teams, Knowledge is the accumulated business know-how that is brought into the project through people and other organizational artefacts and the Intellectual capital is the sum product of all the ingredients that

[32] Senge P, 1990
[33] Waterson et al, 1997
[34] Becerra-Fernandez et al, 2004
[35] Laudon KC & Laudon JP, 2006.
[36] Florida R, 2002

are encapsulated into the final product – the software application. Blackler[37] takes this further indicating that software development is knowledge intensive and as such organizations depend on knowledge workers to be able to deliver software applications.

Considering the fact that offshore outsourcing involves splitting the development teams across geographic boundaries, the challenge is how to externalize much of the tacit knowledge resident within local teams and transport it offshore in a manner that allows it to be understood and utilized successfully and whether there is a framework that could assist organizations to bridge this gap . The fact that some of the knowledge is localized and therefore cannot simply be transferred without providing context adds another dimension. Cockburn[38] raises an important point when he questions the cost-effectiveness of distributed development. He argues that project costs increase in proportion to the time it takes for the development teams to understand each other. Combining Blackler's point on the knowledge intensity of software development, Nonaka's views on the challenges of externalizing tacit knowledge and Cockburn's assertions around the cost-effectiveness of distributed development given knowledge transfer to cross border teams, one can see the challenges involved in organizations deciding to take their development processes offshore and the inherent failures that they have to mitigate for the operation to be economically viable. Maier[39] asserts that learning, no matter individual or organizational, cannot take place without memory, he describes memory as a system capable of storing things perceived, experienced or self-constructed beyond the duration of actual occurrence and of retrieving them at a later point in time. If one juxtaposes this statement with Becerra's assertions it becomes clear that apart from providing business functionality, software applications also form part of organizational memory. From Boisot we noted that an organization's products give a view of the organization's level of knowledge and sophistication[40]. In Information Technology enabled service organizations such as the Financial Services sector, the services offered to clients are heavily underpinned by IT applications, and as such it can be argued that these IT applications are an embodiment of an organization's knowledge and business processes. Extending Boisot's view, one can talk of the software

[37] Blackler F, 1995
[38] Cockburn A, 2002
[39] Maier R, 2004
[40] Boisot M, 1999

applications as being the "physical substrate" for organizational knowledge. Cohen and Young[41] warn that even if service inputs and outputs are quantifiable, a nondomestic operation may require a great deal of knowledge transfer before it can deliver the services intended. In these cases geographic, cultural and language differences can complicate knowledge transfer and overcome any cost savings derived from the lower wages.

2.6.3 Outsourcing as a business strategy

It has been indicated before that outsourcing is on the increase; while the manner and form in which it is implemented might change going forward; there is certainly no closing the Pandora box on outsourcing. This research would not be complete without delving into the strategic motivations for organizations to outsource their applications development to a third party located in a country thousands of kilometres away sometimes exhibiting huge cultural and language differences. From a neo-classical economic point of view, at its fundamental, business is about maximizing profits and minimizing costs and for outsourcing to earn its place in the "business table" it needs to display some or all of these fundamental characteristics. Lewis[42] describes outsourcing as hiring of an external business entity to provide the capability and services needed to augment; compliment or provide additional support for the organization's core business activities along with some or all of the management effort required to direct and maintain those services over time. As indicated above, outsourcing is a supportive strategy meant to free an organization to concentrate on its core business. Outsourcing means different things to different people as St.Amant[43] indicates that some organizations see outsourcing as a strategy for remaining competitive, others view it as a strategy for entering into the global economy while others adopt a dim view and see outsourcing as relocating jobs overseas and closing down local businesses thereby creating unemployment and hardship for the locals. From the above it can be seen that reasons for outsourcing vary from organization to organization.

[41] Cohen L & Young A, 2006
[42] Lewis M, 2006
[43] St.Amant K, 2010

A study by PMP research[44] identified the following key reasons for organizations to outsourcing some of their operations, namely:-

- Reduce operational costs
- Improve service levels
- Focus on core business
- Improve responsiveness to changing business
- Access to specialist skills
- Reduce risk
- Reduce capital costs
- Migrate to new technologies
- Access to advanced technologies
- Align IT with business goals

A cost justification for software development outsourcing measures the cost of developing the software internally against using a vendor, and the decision is made based on the cost differential, however the PMP Research report above indicates the strategic nature that outsource relationships have developed to; as a result cost is but one element of consideration in the outsource decision. Gilley and Rasheed[45] concur with the above point when they emphasize that outsourcing cannot be defined simply in terms of procurement activities as this would not capture the strategic nature of the relationship.

2.7 Conclusion

The introductory chapter has tried to paint a picture of the richness of the software development subject and how the research study intends to explore the outsourcing of software development activities.

[44] PMP Research, 2004

[45] Gilley MK & Rasheed A, 2000

The rationale supporting the research study was put forward and the methodology to be followed in researching the topic motivated. Software development is multi-faceted and this chapter gave a view of how these facets will be covered while narrowing the focus on the topic of knowledge management. A comprehensive literature research was done in order to gain a better understanding of the different views around software outsourcing; the literature research was broken up into several facets so as to explore each angle of the software outsourcing discourse and to discuss leading views that emerge out of the analysis. The following chapter takes the discussion further by looking into the software development process and investigating its knowledge generation capabilities.

Chapter 3
Knowledge creation during the software development process

3.1 Introduction

In the preceding chapter we presented the research concepts used to underpin this research report as well as the business case for undertaking such a research. The software outsourcing rationale was debated through an extensive literature research and varying arguments on software development outsourcing were explored. This chapter isolates the software development process, splitting it into its various stages; the aim being to track the creation of knowledge in each of these stages in order to highlight the criticality of knowledge in the software development process and the need to manage such knowledge.

3.2 A High level view of the software development process

The end product of an enterprise software development effort is the creation of a software system that provides certain business functionality or the maintenance and enhancement of an existing system to meet changing business needs. This "knowledge intensive[46]" development effort is normally driven through a process or methodology that the organization adopts and refines over time, this methodology is referred to as the *Software Development Lifecycle* (SDLC). The SDLC methodology used by an organization is often varied and modified to suit the circumstances and conditions of the software product being developed; this experimentation[47] allows the development team to gain new insights and understanding of the business, tools and the different nuances of the process involved in developing a new software product. As the team involved in the software development process becomes more proficient in the use of the methodology and associated toolset, tacit knowledge begins to develop within the team members on the best ways and nuances of how the tools and methodology can be leveraged for best results.

[46] Blackler F, 1995

[47] Basili V & Rombach HD, 1991

The technical team through its constant contact with business domain experts begin to understand the intricacies of the business value chain and are better placed to convert business requirements into an appropriate technical solution. Detailed knowledge is gained about the software product being developed allowing for its easy maintenance and enhancement at a later stage. While some of this knowledge will be explicitly codified and captured in the department's process documentation repository as part of lessons learnt, the bulk of it is not captured and therefore continues to exist as tacit knowledge that resides only in the heads of team members involved in the project delivery[48]. This tacit knowledge is critical as it allows the organization to develop, deploy and maintain systems more efficiently giving it advantage over competition. The reuse of this knowledge also allows the organization to deploy and enhance systems more quickly and efficiently allowing it to gain advantage in its attempt to be first in the market.

It is pertinent to note that knowledge management generally takes a back seat in software development projects as the focus is largely on delivering a solution on time, within cost, at the expected timeframe with the necessary functionality. Focusing only on solution delivery and not tracking and accounting for knowledge production results in organizations missing the opportunity to leverage and benefit from produced knowledge.. Often software development project make use of external expertise who bring immense subject matter knowledge into the project, upon departure of these experts; such knowledge is lost if the organization does not make plans to retain it.

3.3 Tracing knowledge creation along the SDLC

The software development lifecycle (SDLC) is a mature engineering discipline that continues to evolve influenced by newer technologies and business needs for better, cheaper applications that are quick to deploy.

[48] Henninger S, 1997

There are various approaches to developing software, at the very least the Software Development Lifecycle process is made up of the following steps:-

- Requirements gathering, Analysis and Specification
- Solution design
- Solution construction
- Testing
- Deployment
- Support and Maintenance

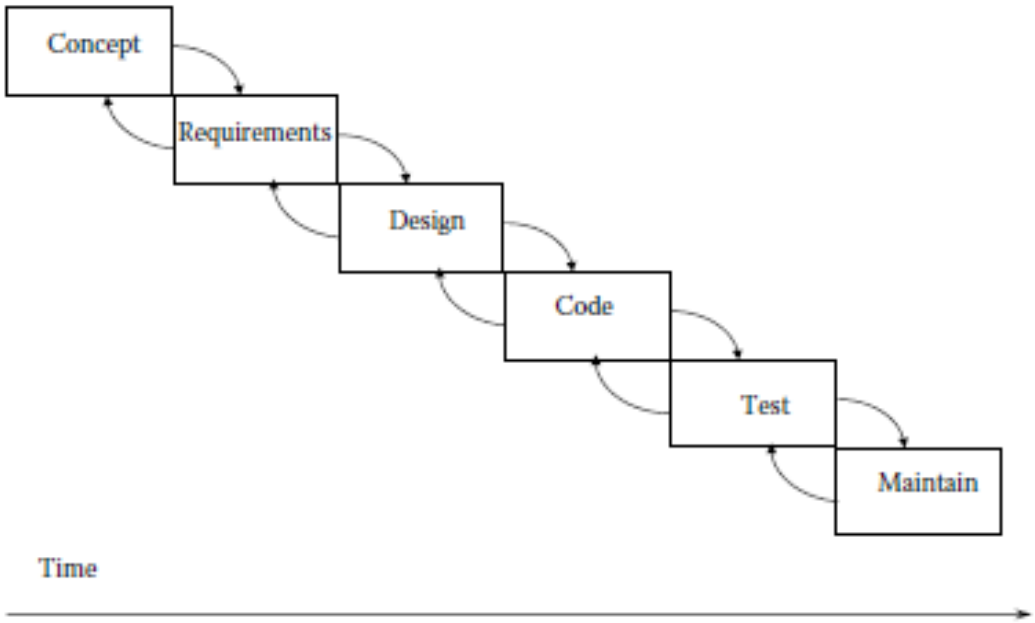


Figure 3.1 - Stages of software development represented through a Waterfall lifecycle model *(Source: Laplante PA, 2007)*

Supporting the Software Development Lifecycle are the following disciplines:-

- Project management – a function that binds and coordinates the software development activities and primary role players into a coherent delivery unit.
- Processes, Methods and Tools – The accelerators that seek to ensure uniformity across all software delivery efforts the organization undertakes.

These activities are not mutually exclusive but instead complement each other to produce a solution that meets business requirements. The SDLC can be said to be Waterfall in which case its stages follow each other in sequence or Iterative (Incremental) in which case the stages run concurrent to a certain extent. Which method is chosen depends on the business circumstances and the solution being delivered.

While the SDLC governs the delivery framework, the process of producing software cannot be reduced to a simple instruction following exercise, team members constantly make decisions regarding their approach to software development issues and as such the approach will differ from team member to team member within the boundaries of the framework depending on his experience and background.

3.3.1 Project management knowledge

Every project follows a standard set of guidelines which have been proven over time to deliver results; these are encapsulated as best practice in the Project Management Body of Knowledge[49] (PMBOK). Figure 3.2 below shows the main phases of a typical project as defined within PMBOK and the process flows within each phase.

[49] Project Management Institute, 1987

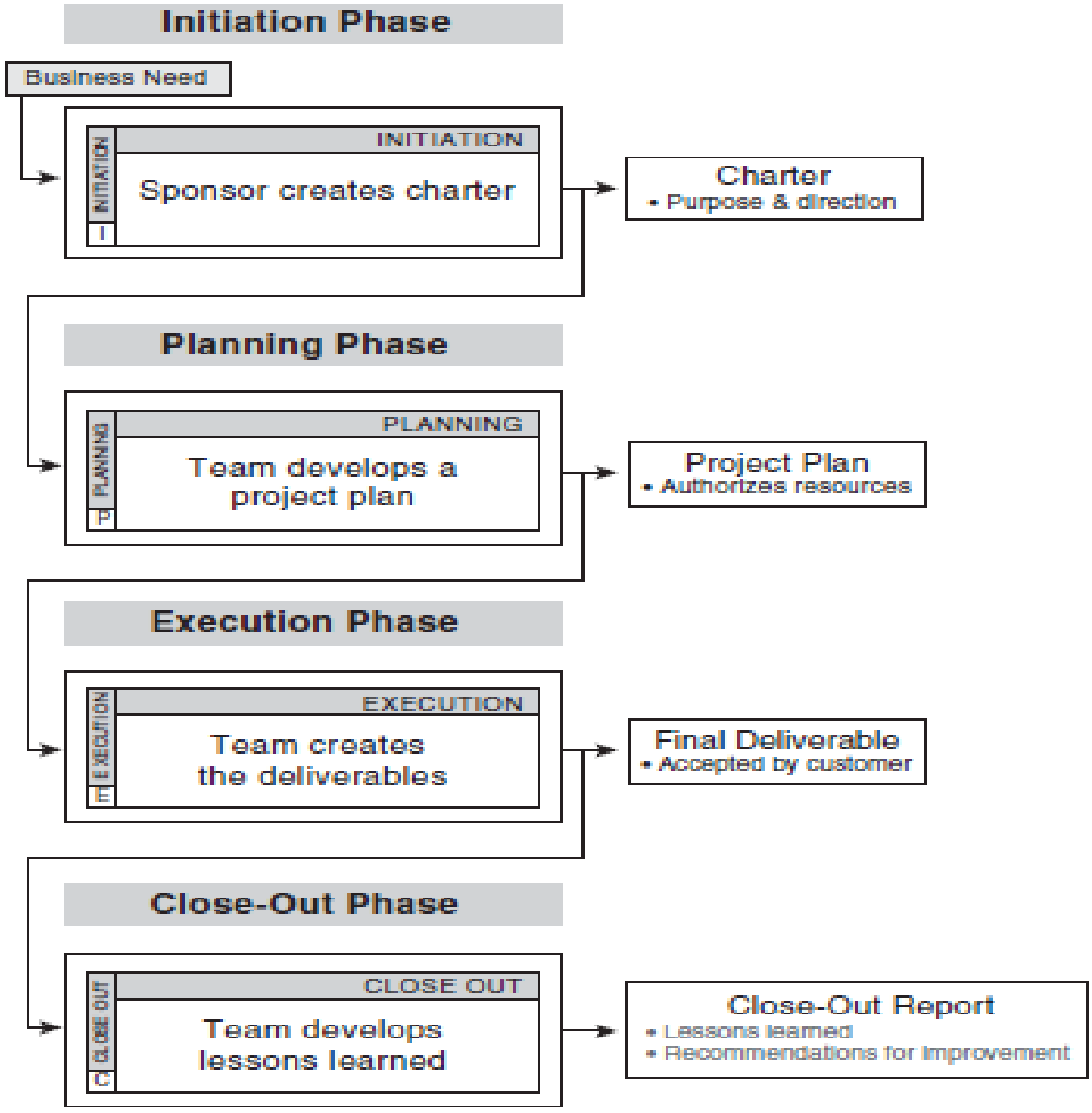


Figure 3.2 - Process flow within project phases

(Source: Martin P & Tate K, 2001)

With only 32% of software projects deemed to be a success[50], software development is still a highly unpredictable exercise. Software project success is largely dependent on management discipline and the maturity of organizational processes rather than technology advances.[51]

[50] Standish Group, 2009

[51] Jones C, 1996

The above statements indicate the importance of knowledge management in project success as management discipline and process maturity is shaped by knowledge gathered from past experiences. To start with a project, a project manager needs to have some basic fundamentals regarding the project to be undertaken; these include:-

- The nature of the product to be developed
- The constraints placed by the organization on the development initiative regarding Scope, Time, Quality and Budget.
- The type of resources needed to undertake the project and the level of knowledge required to be able to succeed.

As the project manager goes through the planning process, he brings with him tacit knowledge on similar projects he has delivered before, he also engages with project stakeholders to contextualize this knowledge to the problem at hand. This knowledge is mostly tacit and resides mainly in the project manager's head. A project delivery methodology to be used to drive the project often has to be agreed upfront. Often this methodology is a variation of the standard PMBOK and SDLC processes shown in Figure 3.1 and Figure 3.2 above; this master document is codified and filed as part of the project deliverables. Sometimes because of the peculiarities of the project, a specific methodology needs to be adopted to ensure success of the project. New knowledge is created in the process of discussing and refining an existing methodology and while the new methodology will be documented as an artefact; the nuances that gave birth to its existence might not all be captured and as a result will remain tacit in the heads of those involved.

3.3.2 Software engineering processes, methods and tools

Software development is a complex engineering undertaking, software practitioners utilize Tools, Methods and Processes to leverage off existing knowledge and to fast track product delivery.

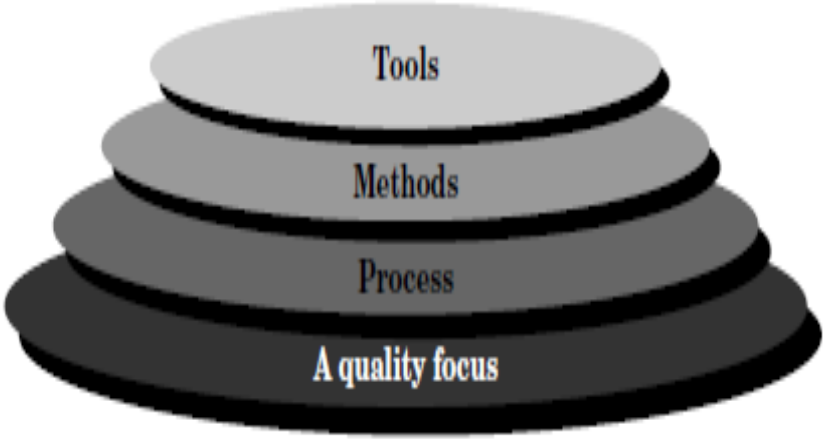


Figure 3.3 - Software Engineering Layers

(Source: Pressman R, 2001)

Shown in Figure 3.3 above are software delivery layers containing embedded knowledge meant to guide software delivery activities ensuring that a uniform way of working is followed and that the deliverables conform to a prescribed quality standard. The above layers provide the following support to the software development process:-

- Processes define a framework that must be followed for a successful delivery of software solutions. This framework is based on research and knowledge gathered from previous experiences which is then encapsulated in some best practice that the organization has adopted.
- Methods provide the know-how and the how-to for building software.
- Tools provide automated support for the Methods and Processes.
- Quality is the foundation underpinning all the layers ensuring that the framework, its processes, methods and supporting tools produce a product of high standard.

Proficiency in the use of Processes, Methods and Tools increases with reuse and this knowledge is evidenced by the quality of project deliverables produced and the knowledge accumulated by those who participate in such projects.

Level	*Continuous Representation Capability Levels*	*Staged Representation Maturity Levels*
Level 0	Incomplete	N/A
Level 1	Performed	Initial
Level 2	Managed	Managed
Level 3	Defined	Defined
Level 4	Quantitatively Managed	Quantitatively Managed
Level 5	Optimizing	Optimizing

Table 3.1 - Software Engineering Institute (SEI) Capability Maturity Model levels

(Source: Software Engineering Institute. 2006)

Organizations use the Software Engineering Institute's Capability Maturity Model (CMM) to measure the level of standardization and proficiency achieved in their software delivery processes, where level 1 signifies chaos and inconsistency of delivery processes while level 5 is the highest maturity level indicating documented, repeatable and highly optimized processes. Table 3.1 above shows the different levels of the CMM scale, organizations at CMM level 5 have mature processes and as such their knowledge management activities are likely to be at an advanced level.

3.3.3 Requirements gathering, analysis and specification

Key to the development of a new system or enhancement of an existing system is the understanding of a user's requirements. Requirements represent the benefit the user expects to derive from the delivered system, requirements gathering therefore engage the user in an attempt to understand the:-

- Objectives of the system and the business problem to be solved
- Scope of the solution.
- Constraints to be observed when developing the system and while operating it.

During the elicitation and analysis of business requirements, IT team members get to query business stakeholders on their requirements, process flows and strategic direction going forward. These interactive sessions allow the project to gain valuable insight into the functioning of business from domain experts. Sometimes this is the only time that such knowledge is codified or updated; therefore the software development process, besides delivering a new system, allows the business to capture tacit knowledge regarding its processes and business rules and convert this into sharable artefacts. The documented system then becomes part of the company's knowledge repository. Because knowledge is captured in the context of writing specifications for the building or maintenance of a particular system, much of the valuable knowledge shared can be deemed out of context and therefore remains tacit and is transferred to the IT analyst without being codified. The Software Requirements Specification (SRS), an end product of the requirements gathering activity, is an artefact that details user requirements and system behavioural expectations. When the SRS is thoroughly and properly completed it contains enough information to allow the rest of the project team to understand and build the system in line with user expectations. The risk of business and project team members leaving in the middle of project activities is mitigated as all the information required to build the system is contained in the SRS. Knowledge about the criticality of the system and constraints in terms of scope, budget, time and level of quality under which it should be built is communicated and captured. External vendors bidding to be part of the project can use the requirement document to gain enough knowledge about the organization and understand its workings well enough to propose a solution that best meets the needs of the users. Detailed capturing of requirements also mitigate against cultural and linguistic differences that might arise from a diverse stakeholders community.

Requirements take different forms which include[52] :-

- **User requirements.**

  These reflect user needs and how they will view the system from the perspective of input and outputs.

[52] Turner J, 2007

- **Functional requirements.**
  These express the functionality the system should possess and the general operations of the system
- **Derived requirements.**
  These are the requirements that are not necessarily stated but are implied as part of the package.
- **Technical requirements.**
  These describe the physical attributes of the system to be built.

All these are different sides of the same coin and are intended to give context to the character of the system to be built.

3.3.4 System design

From a technical point of view system design converts the specifications into a practical design that can be implemented by system developers. The aim is to decompose the requirements into Data, Architecture, Interface and Component designs. Business rules, Controls and Process flows are stipulated in the design, this forms part of the business knowledge that is embedded into the solution. The solution then forms part of the organization's knowledge assets.

The designer has a complete understanding of both the knowledge embedded in the solution as well as knowledge about the functioning of the solution (Meta knowledge – i.e. knowledge about knowledge). This knowledge, accumulated during the development of the solution, allows the designer to enhance the solution in future with relative ease and to build similar solutions much quicker giving the organization the edge over competition. System designs contain enough knowledge to allow future teams to understand how the solution hangs together and therefore make future maintenance and enhancement relatively easy. System designs are also the basis for future systems, new and innovative solutions are based on knowledge gleaned from existing designs.

3.3.5 System construction

When constructing a system, developers use frameworks and tools as determined by the organization's technical standards. Depending on functionality being built and commonality of

projects, it is possible to reuse artefacts built by other projects and therefore leverage off existing knowledge. This has the effect of accelerating project delivery, reducing costs and improving product quality. If knowledge management discipline is not followed these reuse opportunities are not spotted and therefore rework occurs resulting in cost and effort duplication.

Base frameworks such as Sun's Java2 Enterprise Edition (J2EE) and Microsoft's dot.Net provide prebuilt environments on top of which business applications can be built. Higher level frameworks such as Spring take this concept further by providing pattern solutions that hide technology complexities and enforce best practice methods through prebuilt, reusable functional components. Many organizations are investing in these technologies to streamline their development operations and improve productivity of their technical staff. Some organizations further customize these frameworks to create development environments that are proprietary to their needs hence further abstracting technology complexities and presenting to their technical staff only the relevant information they need to do their job. Development tools come with enough prebuilt knowledge that organizations do not need to employ scarce, highly paid experts to build sophisticated business applications. Developers also enhance their knowledge and understanding on the use and application of tools and frameworks thereby improving their proficiency and effectiveness. With each project iteration; the organization's process maturity is lifted as indicated earlier from the CMM model; this in turn improves its knowledge management competency. During project delivery, organizations usually collaborate with external vendors who are experts in the tools used; knowledge management ensures that internal skills are increased thru sharing and harvesting of this knowledge.

3.3.6 System testing

Before a system is tested, test plans and test cases are prepared in order to verify the finished product against set requirements. The test analyst needs to have enough knowledge about the business rules to anticipate system behaviour and test for the necessary conditions. As conditions are tested, certain discoveries are made about the business rules and new exception rules are developed. This new knowledge is then fed back into the system as part of the revised specification; documentation (knowledge artefacts) is updated accordingly. Some of this knowledge only remains with a few individual project members and does not find its way to the

wider business community. During system testing, the project team tests the system against test cases derived from system requirements. If the test cases are not comprehensive, errors could creep into the system and compromise the integrity of the business. Knowledgeable project resources would use their understanding of the business and prevent such oversights.

3.3.7 System deployment

System deployment is the process of transitioning the built and tested solution to the production environment so that the new functionality becomes available to the organization's user community. The deployment process is largely a coordination process and because of audit requirements and regulations governing separation of duties, the development team is forced to hand over to a separate deployment team to transition the solution into production. This separation aids in spreading the acquired knowledge to a wider community than the few involved in the building process, ensuring that any flaws in the documentation are discovered, thus enhancing the overall quality of the knowledge assets generated. Organizations are opting to automate the deployment process wherein all the process and rules involved in deployment are encapsulated into automation scripts that handle everything from packaging the required software modules, do version control and security checks to deploying the software into the production environment for consumption by the organization's users.

3.3.8 System support and maintenance

A software application is like a living organism; when in production it is likely to develop issues that will need attention, these include inter alia functionality that does not perform quite as planned and performance that is not in line with business requirements. New business requirements also imply that the solution will need to be tweaked to cater for changing business needs. Documentation and other knowledge artefacts created during the delivery process play a vital role in aiding the understanding of how the system functions within its business domain context and in planning for and making the necessary changes to adapt to a changing environment. The extent to which the system is documented and the quality of the knowledge transfer during the build process often becomes apparent during the Support and Maintenance stage, more especially if the team maintaining the solution is different from the original delivery team that created the solution and the original team has since moved on. Also to ensure

separation of duties as prescribed by audit and governance policies, it is often required that the team building the solution be different from the maintenance team, making knowledge management even more essential for the longevity of the solution.

One of the biggest challenges of outsourcing software development is how to transfer the large pool of tacit knowledge from internal resources to the vendor in the short space of time to enable the vendor to deliver a service that is similar or better than internal resources without compromising the organization's intellectual property and where necessary to transition the service back, retaining all the knowledge accumulated during the engagement period. This challenge becomes even more daunting when the outsource engagement is to an offshore vendor. A well-executed project can be easily undermined by an inadequate knowledge management strategy.

3.3.9 Configuration management

The software product continues to evolve throughout the development lifecycle as users modify their requirements and new functionality is added. Configuration management keeps track of all the facets of a software product providing a repository that contains detailed information about the structure of the product, the changes implemented and status of the current version.

3.3.10 The software development ecosystem

As indicated above, software development enlists the participation of a number of role players, each with specific expertise to do specific tasks that contribute towards the bigger whole. Software development is also a disciplined activity with all its processes governed by standards, quality frameworks, and best practices.

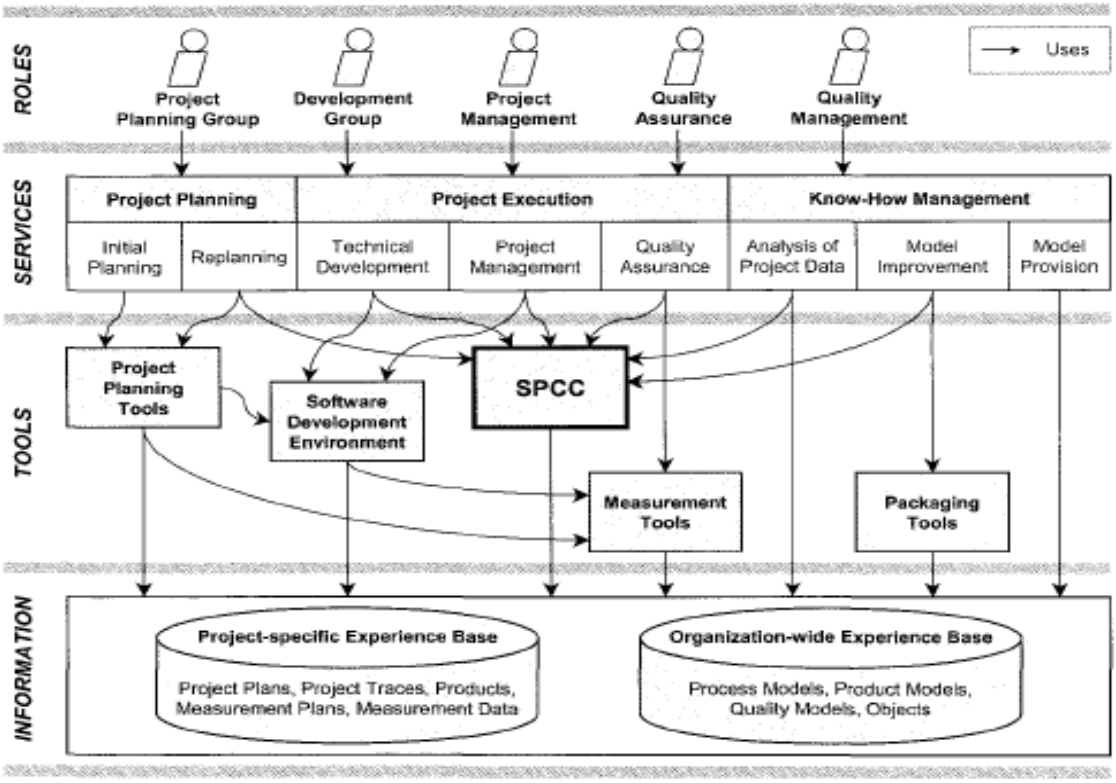


Figure 3.4 - The software development ecosystem

(Source: Munch J & Heidrich J, 2005)

Figure 3.4 Illustrates the roles involved, services provided by each of the role players, the tools used in the creation of a software product and management of the activities involved as well as the information gathered and knowledge processed. The SPCC (Software Project Control Centre) has the responsibility of ensuring that project objectives are met by monitoring and measuring progress as the project continues, it provides support to the project control activities. It assists the project management team in decision making by extracting, interpreting and visualizing data in comparison to base plans and information from previous projects. Information collected resides in the projects repository and is constantly mined to pick up trends, patterns and variances which help in populating the knowledge repository. The knowledge database is a know-how and experience repository which becomes a reference point for future projects. All project services namely Project planning, Project execution and Know-how management use the knowledge repository as a reference point for their activities. Tools are used to access, mine and analyse the know-how, in the process creating new knowledge which further enriches the repository.

3.4 Knowledge assets within a software development project

As demonstrated above, the software development life cycle gives rise to various forms of knowledge assets both tacit and explicit, organizations therefore need to be aware of these assets and formulate strategies for securing them. While the systems being developed might be the primary goal of an SDLC effort, it is wise for an organization to think of the knowledge assets generated by the processes as economic goods in their own right. Once organizations start to leverage this knowledge, they develop competencies which might give them the edge over competition. Knowledge gives an organization the ability to economize on the use of physical resources namely space, time and energy[53] as a result with the right knowledge, application systems are developed faster, cheaper and with less effort :-

- **Faster** - Organizations leverage their accumulated knowledge to make the right choices quicker.
- **Cheaper** – Time is money, therefore building systems quicker saves time and therefore money.
- **Less effort** - The ability to reuse knowledge components from previous initiatives results in reduction in development effort.

Extending Becerra-Fernandez et al[54] assertion to the software development environment one can argue that knowledge created in the SDLC process resides in the artefacts created, the people involved in the creation of these artefacts and the organizational structures participating in the process. One of the key responsibilities of a successful project would therefore be to account for all knowledge assets created in the life of that project. Perusing project management literature, one gets the sense that knowledge management is unfortunately not yet acknowledged as one of the qualities required for projects success; focus still remains on the constraints triangle of time, quality and cost[55].

[53] Boisot M, 1999
[54] Becerra-Fernandez I et al, 2004
[55] Application Executive Council, 2007

3.4.1 Explicit knowledge within the Software Development Lifecycle

Explicit knowledge is codified knowledge, it is knowledge that has been expressed in words and numbers, such knowledge can be shared formally and systematically in various forms such as data, specifications, manuals, computer programs and audio visual material[56].

In the Software Development Lifecycle, explicit knowledge is captured in such artefacts as:-

- Business requirements documents
- System specifications
- Solution architectures
- System designs
- Reference frameworks
- Development standards
- Project and Software delivery methodologies
- Risk and Issues logs
- Governance protocols and engagement minutes

We will now look at each of the above artefacts to understand the knowledge qualities they possess.

- **Business Requirements Document (BRD)**

  The business requirements documents is the starting reference for a software development project, it provides a high level understanding of what the organization aspires to become, gleaned from its strategy document and the objectives that will contribute towards the attainment of this vision[57]. It contains in detail the functionality required by the business, the context and rationale of the system as well as scope of the business areas targeted by the system. The business requirement document communicates the business world to the IT world and from it a lot of assumptions are gleaned which later form part of the new system. Since the business requirement document is compiled in conjunction with business domain experts, it either updates existing knowledge on business processes or documents the business areas from scratch. On both occasions up to date knowledge is captured on the functioning of the business area concerned. In many instances the business requirements

[56] Becerra-Fernandez I et al, 2004

[57] Application Executive Council, 2007

document becomes critical as it is the only place where such knowledge about the business and the system context is captured. The business requirements document therefore secures the captured knowledge for in the event where both the business domain expert and the analyst involved are no longer available. The business requirements document becomes critical in outsourced SDLC engagements as the vendor normally does not have the context of the client's business and as such depends on this document to form a picture of the business requirements and to brief the development team on business needs in focus. Apart from documenting business requirements; the business requirements document therefore transfers knowledge to a variety of stakeholders involved in the creation of a business solution.

- **System Requirements Specification (SRS)**

  System requirements describe services to be provided by the system and the operational constraints under which the system will function[58]. This document describes the characteristics of the system which in turn become condition for its acceptance by business. It is a technical interpretation of the business requirements and is used by the technical team downstream to construct the system. From the SRS document designers and system architects make decisions on how the system should be put together to best meet user requirements, this demonstrates the cascading of knowledge from the business users down the delivery chain to the technical team and the influence of such knowledge in the decisions taken.

- **Architecture and Design documents**

  Architecture and design documents specify how the system should be put together to best deliver on the business's requirements. While this document guides the construction team during delivery, its importance becomes apparent later when the delivery team has moved on and the support team needs to maintain the solution going forward. The architecture and design documents contain knowledge about the components that make up the system, how these components interact together and how the system interfaces with its intended users and other systems in general. The knowledge shared by business during the construction of the system and the expertise of the technical teams involved in construction of the solution is

[58] Sommerville I, 2007

embedded in the architecture document. Any change or enhancement to the system cannot take place without reference to the architecture and design documents. Architecture documents normally incorporate the IT strategy of the organization and best practices that the delivery team has adopted.

- **Reference Frameworks and Delivery standards**

  The construction of a software system involves many people of varying expertise, experiences and approaches. Organizations introduce standards to harmonize the various approaches likely to develop between team members and to ensure that the processes are repeatable regardless of the diversity of team members involved. Repeatability breeds proficiency and this in turn increases the level of maturity in an organization.

  The process of software development has been around for a long time and a lot of knowledge has been accumulated over the years. To leverage off this knowledge; organizations make use of Reference Architectures which are generic solutions that can be adapted and reused to fast track software development effort. Reference Architectures could either be a set of guidelines on how to approach a particular problem, standard pre-built components that organizations can use to solve standard problems or internal solutions built by senior members of the technical team to standardize and fast track certain operations. Reference architectures are also used to abstract the complexities of underlying technologies allowing organizations to tackle challenging requirements with less experience.

- **Risk and Issues log**

  Risk and issues log are a source of information on the risks and issues the delivery team had to face, how these were resolved and the extent to which they were mitigated. The actions taken to mitigate these constitute learning by the team and as such knowledge that is created for future reference. Unmitigated risks are a potential source of trouble and require the organization to equip itself with the right knowledge to deal with the consequences should the risk materialize. The challenge in outsourced development initiatives is ensuring that the knowledge accumulated in mitigating risks is transferred to the client and that unmitigated risks are made visible with enough information on how to resolve them.

- **Governance protocols and engagement minutes**

  Software delivery projects have formal governance structures where status is shared, decisions taken and direction is given. Key stakeholders and Sponsors use these forums to set priorities and to offer insight on strategic issues. The governance structure by its mere existence becomes a source of knowledge and the coming together of experts to guide the project gives rise to new knowledge which needs to be captured. Minutes of project meetings act as a record of decisions taken and insights generated. Project sponsors are fairly senior members of the organizations and are not normally accessible; the governance forums are the few areas where their insights and knowledge of the organization can be accessed.

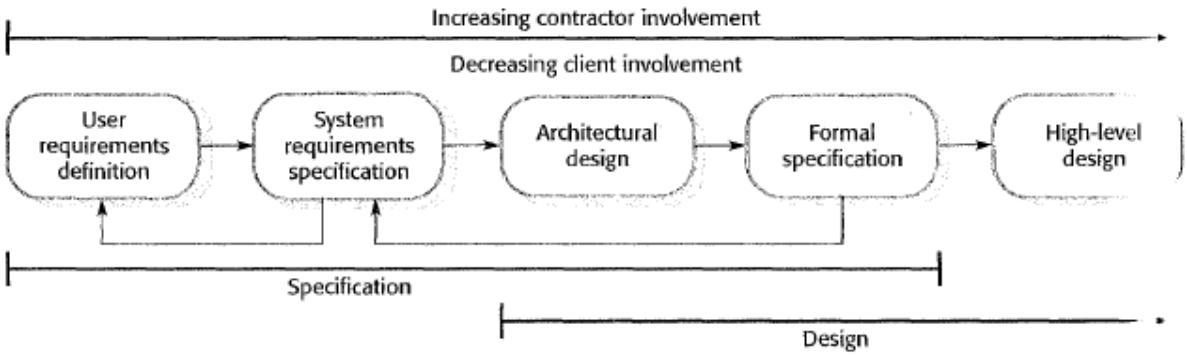


Figure 3.5 - User participation levels in System specification and Design stages

(Source: Sommerville I, 2007)

Illustrated in Figure 3.5 is the decreasing involvement in user participation as the project proceeds down the delivery timeline highlighting the need for effective knowledge management and transfer throughout the software delivery lifecycle.

3.4.2 Tacit knowledge in the Software Development Lifecycle

Unlike the explicit, articulated knowledge discussed above tacit knowledge is intangible knowledge that is not codified; it resides in the hearts and minds of the many team members involved in the delivery of a software solution. This knowledge is difficult to express and

formalize and therefore difficult to share[59]. Members of the delivery team acquire this knowledge as experience by virtue of being members of the team and being exposed to the various discussions taking place and also by the experiences they go through during the delivery of the software solution. In outsourced development engagements the vendor will sit with business domain experts and the organization's technical experts to extract and codify some of the tacit knowledge the business experts possess concerning the organization and its systems, after that the vendor will move offshore to proceed with the construction of the new system. Because of the separation of the teams the client does not participate fully in the knowledge generating interactions as a result the learnings that accrue from such interactions are lopsided in favour of the vendor as the client does not get to harvest knowledge from the vendor's experienced team, the vendor on the other hand will use the acquired knowledge from client business and technical experts to enhance its own and utilize it in future engagements.

3.4.3 Challenges created by the accumulation of tacit knowledge

Below are some of the organizational challenges posed by the unintended accumulation of tacit knowledge.

- Organizations do not know what knowledge they have and where it resides, as a result they are unable to leverage such knowledge effectively.
- For its true value to be realized knowledge needs to be shared, cultural and team dynamics can however be the barrier to a successfully transfer of knowledge resulting in knowledge being concentrated in a few individuals.
- With the turnover of employees being high in the IT sector, it is easy for organizations to lose knowledge without being aware of such loss and the potential impact and exposure created.
- For some technology organizations work practices are not documented but are rather driven by technology champions who take it upon themselves to coordinate activities, this generates dependency and this tacit knowledge is lost when the champion is run over by the proverbial bus.

[59] Becerra-Fernandez I et al, 2004

- IT team members through their constant contact with business develop a deep understanding of the business areas they support and the make-up of the systems used in these areas, this integration knowledge is generally not codified, neither in business nor in IT.
- Information Technology is ever changing, while organizations might initially have had elaborate documentation, this soon becomes obsolete due to changes and as such the explicit knowledge becomes irrelevant and therefore useless, this creates complacency as the organization continues to believe that all its knowledge is still intact.

While it is accepted that not all tacit knowledge will be codified, it is incumbent upon software development organizations to ensure that much of the tacit knowledge is identified, codified and made explicit so that it can be better leveraged for the good of the organization.

3.5 Codification and abstraction in knowledge sharing

Organizations are constantly faced with choices between complexity absorption and complexity reduction. Complexity absorption leads to a steady accumulation of tacit knowledge[60] which exposes the organization to the risk of losing knowledge; above all as indicated above, tacit knowledge cannot be so easily shared. While the obvious choice would be a move towards complexity reduction through codification and standardization, care needs to be taken when adopting this approach as it could be perceived as excessive control and anti-innovation. The knowledge worker values a sense of freedom and space in the workplace to exercise his craft and any perception of restrictions is normally met with agitation followed by resignation. Knowledge codification therefore has to be sold as a value proposition that stands to benefit both the employer and the employee; this could generally be done through a carrot and stick process of laying the rules and giving rewards for adherence.

3.5.1 Knowledge codification

Knowledge codification is a process of capturing and documenting knowledge using a format that can be understood by the target audience[61]. Not all information constitutes knowledge and not all knowledge is worth the effort of being codified, some knowledge is plain common sense

[60] Boisot M, 1999
[61] Boisot M, 1999

and as such does not require codification as this does not add value. Organizations therefore have to constantly glean from the pool of available knowledge the elements that are worth the effort of codification.

3.5.2 Knowledge abstraction through software development frameworks

Abstraction is the process of extracting from available knowledge a generalized version that is capable of being applied to a variety of scenarios, put differently, it is generalizing the application of available knowledge and insights to a wider range of situations[62]. Through abstraction organizations are able to simplify complex concepts breaking these down to levels that the average team member can grasp and utilize. Development frameworks are a classical example of utilizing abstraction in the software engineering domain, using frameworks organization abstract from complex development environments only the parts relevant to their environments and expose these to their developer thus fast tracking the assimilation of new knowledge and expediting the learning process.

When outsourcing to software development vendors, clients have the benefit of tapping into many of these frameworks which the vendor has accumulated over time to accelerate delivery. As the client's core business is not software engineering, they would not normally have such knowledge assets and if they do it would normally take them a long time to accumulate and perfect to the level and standard of the vendor's.

3.5.3 Sharing knowledge within a software engineering project

It is easier to share knowledge once codified, organizations can bring on board external vendors and new employees with ease if all the organization's procedures, standards and guidelines are well documented. In an outsourced development environment, this efficient onboarding translates to savings as the vendor spends less time complying with the client internal processes and more time doing the work he was brought in to perform. Learnings from an engagement can be abstracted into generic trends that are applicable to a variety of situations. Through codification these learnings and experience can be spread further than just the core development team and made available to a wider audience of software developer teams thereby increasing the organizations overall capability.

[62] Boisot M, 1999

3.5.4 Mapping knowledge flows using Boisot's I-Space conceptual framework

The Information Space is a conceptual framework developed by Max Boisot to depict the influence of Abstraction, Codification and Diffusion on the behaviour of information flows when knowledge assets are created. The three dimensions represented by the x, y and z axis on the I-Space framework depict Abstraction, Codification and Diffusion respectively

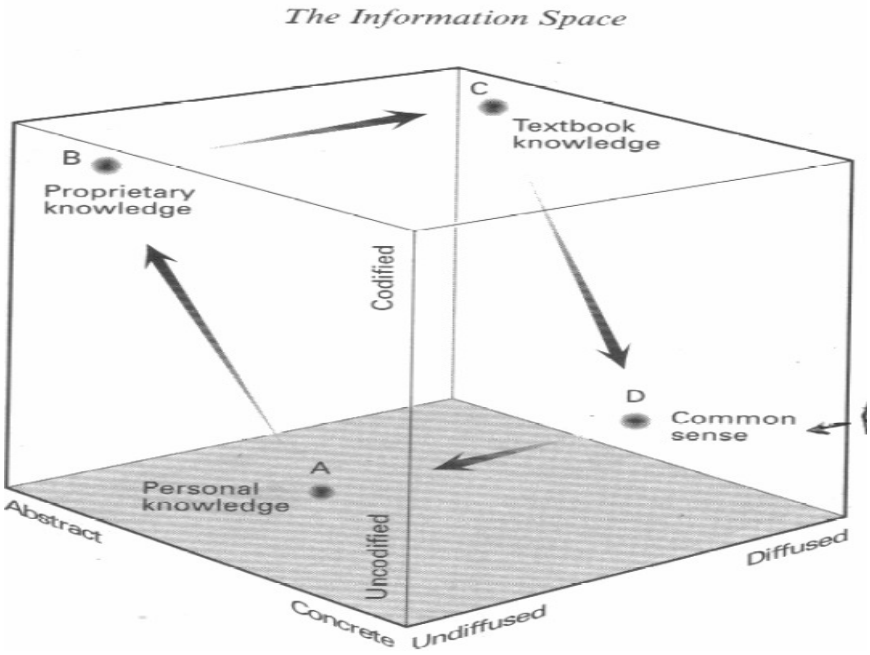


Figure 3.6 - The Information Space conceptual framework.

(Source: Boisot M, 1999)

In Figure 3.6 above is a representation of Max Boisot's Information Space conceptual framework also referred to as the I-Space. For the purpose of definition, the three dimensions of the I-Space framework are:-

- **Abstraction**

  The process of extracting generalized values from available knowledge such that it can be applied in a variety of scenarios

- **Codification**

  The explicit capturing of knowledge into reference-able codes (words and numbers).

- **Diffusion**

  The measure of the availability of information to those who want to use it.

Traversing through the axis of the I-Space from a software developer point of view one can glean the following observations:-

- **Concrete, Uncodified, Undiffused**

  As he goes about his work, a team member would discover a way of utilizing the software development tool that produces better results or resolves an issue thereby making him more productive than before. This knowledge would be concrete as it pertains to a specific issue, it would also be uncodified as it is not documented anywhere, neither in the tool vendor's reference manuals nor the organizations operations documentation. The knowledge would be undiffused as only he knows about the new found way of doing things.

- **Abstract, Codified, Undiffused**

  As he has found this discovery to be valuable he would capture it into his personal PC and constantly refer to it whenever a similar problem arises. As he experiments with the solution, he would suddenly discover that if he changed a few things here and there the solution could be applied (abstracted) to a wider range of uses than the original problem. The knowledge would now be codified, abstracted but still undiffused as only he knows about the discovery.

- **Abstract, Codified, Diffused**

  A project close-out session is normally held at the end of a well-run project, this is where the project team goes through its performance scorecard and review all areas, making recommendations for improvement where necessary. This session is also used for knowledge harvesting where any knowledge generated during the project is shared between team members and made available to the wider software development organization. In this meeting the staff member in question would share the above discovery and make his notes available to a wider audience. This new knowledge will then be evaluated, polished, restructured and packaged for a wider audience. It would be stored in the organization's knowledge repository and published for access by other team members. As a knowledge asset, it would be subject to further refinement by other team members.

- **Concrete, Uncodified, Diffused**

  Sometimes knowledge is socialized to a wider audience without being codified; this concrete knowledge sometimes taken as common sense would be diffused (shared) during informal social gatherings but would not be documented (codified) in an organizations repository. Software vendors generally have better knowledge management processes than their clients and would codify this knowledge for further use without sharing it with the client. With development taking place offshore the likelihood for knowledge sharing becomes limited.

3.6 Conclusion

It is clear from the above evidence that the Software Development Life Cycle functions as a knowledge production line with knowledge being created, captured, shared and utilized not just at the end of the production line but at every stage of the development process. Various forms of knowledge artefacts are consumed as input and new knowledge forms are created in the process of delivering a software solution. What is evident is that a portion of the knowledge created remains tacit to those team members intimately involved in the delivery process, this is a challenge to the process of retaining knowledge. In order to secure the knowledge assets generated, software development life cycle therefore needs to be accompanied by a strong process of collecting, organizing and storage of knowledge. SDLC participants also need to be educated on the knowledge generation capabilities inherent in the process so that they can assist in the retention of such knowledge. The evaluation criteria for frameworks, methods and tools used in SDLC should be expanded to include knowledge management capabilities because the absence of a knowledge management strategy in a software development engagement is an omission that will come back to haunt the organization in time to come.

In the next chapter we look at the strategies and tools that can be employed by organizations in ensuring that knowledge created in outsourced software development engagements is managed to the benefit of the organization.

Chapter 4
Supporting software development with Knowledge Management Systems

4.1 Introduction

Knowledge management involves acquiring, creating, sharing, utilising and storing intellectual assets from the internal and external business environments facilitating an organization to perform successfully[63]. As indicated in the previous chapter, knowledge is a vital ingredient of an organization's software development capabilities; loss of such knowledge can cripple the organization's ability to effectively deliver on its software development strategies and therefore severely affect its operations and profitability. Considering that software applications play a big part in the organization's product or service delivery, the management of knowledge from its software engineering practice becomes key. When organizations decide to outsource their software development initiatives, they are in effect taking a decision to acquiring knowledge from an external party considered to have enough of it to meet the specified needs. All knowledge is initially held by individuals either as tacit knowledge or as personal explicit knowledge[64]; individuals who are involved in software development initiatives are the initial points of knowledge accumulation for an organization and as such need to be managed strategically. Knowledge that is not managed efficiently can results in missed business opportunities as a result of unproductive effort such as rework, reinventing the wheel and bad or ill-informed decisions. Knowledge is being created all the time within an organization and therefore a strategy is needed to facilitate the capture, processing and utilization of this knowledge. Key to knowledge management is interface management which entails the facilitation of interaction and conversation so that knowledge can flow freely. It is through this free flow of knowledge that knowledge can be shared, codified and transformed from tacit to explicit.

[63] Kulanga GK & McCaffer R, 2001
[64] Nonaka I & Takeuchi H, 1995

4.2 Knowledge management processes

Knowledge management processes are the fundamental processes that help in the discovery, capturing sharing and application of knowledge[65]. Figure 4.1 below shows the key processes of knowledge management and the supporting systems that underpin the transformation and coordination of knowledge within an organization.

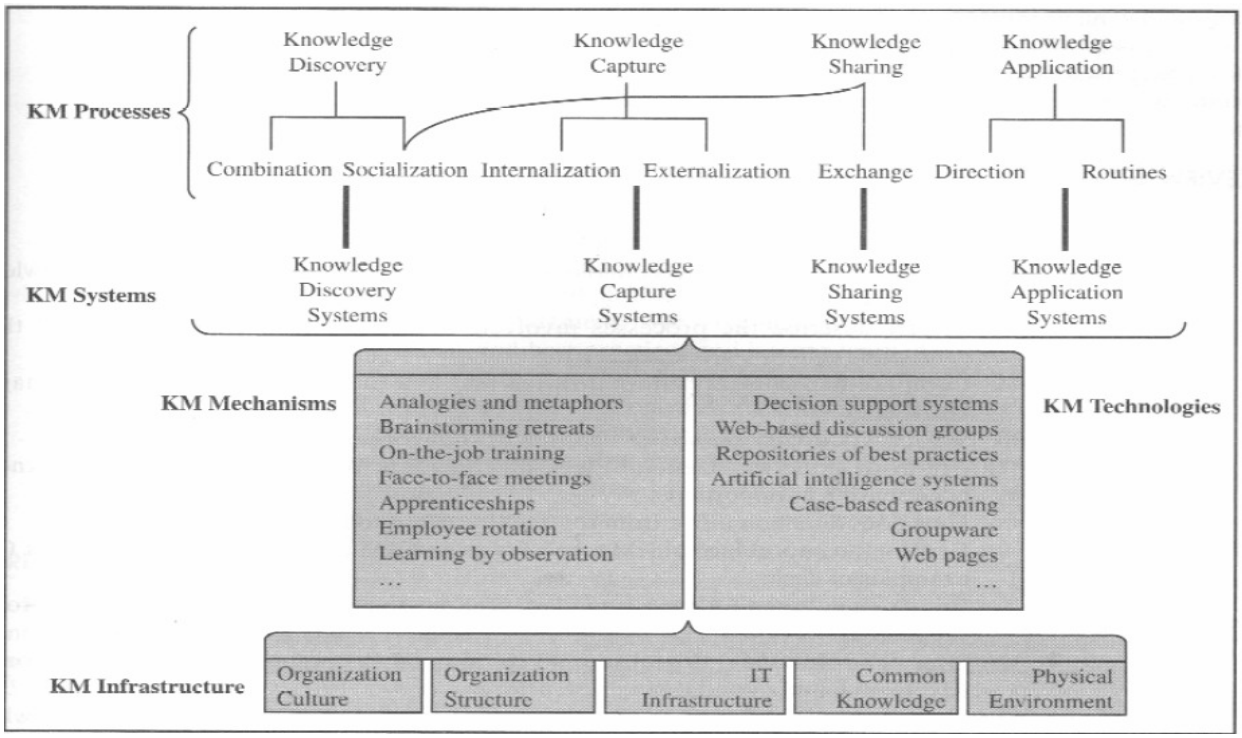


Figure 4.1 - Knowledge management processes and their supporting systems.

(Source: Becerra-Fernandez I et al, 2004)

To get the most out of their knowledge resources, organizations need to implement systems that facilitate a structured manipulation, harvesting and effective management of these resources. It would not yield much benefit for an organization to accumulate knowledge that does not find its way to the relevant people at the right time and therefore does not get utilized in driving forward the organization's strategic objectives. Knowledge processes therefore provide a framework for managing corporate knowledge from cradle to the grave. A developed knowledge management

[65] Becerra-Fernandez I et al, 2004

process framework is an indication of the level of maturity of an organization's knowledge management efforts. Software development is a process driven exercise, for knowledge management to succeed within this environment, its core processes of discovering, capturing, sharing and application of knowledge need to be aligned and complimentary to those of the SDLC.

4.2.1 Knowledge discovery and creation

Organizational knowledge creation involves creating new forms of knowledge or replacing existing content with new tacit or explicit forms of knowledge, this happens through social and collaborative activities as well as through an individual's cognitive reflections[66]. Knowledge is generated throughout the lifecycle of a software development project, this knowledge ranges from tacit knowledge brought in through the experiences of the team members that work on the project to explicit, codified knowledge brought in through the project documentation created, referenced and updated in the process of delivering a solution. There is no "special" session for knowledge generation, team members acquire knowledge in the course of doing their daily project activities, for this reason it is therefore important that an organization has structures in place to capture such knowledge. For a software engineering organization to be able to manage its knowledge it needs to be aware of areas with high knowledge generation potential. Some of the areas with high knowledge generation potential are:-

- **New challenges**

  As project team members go through new challenges of building new software applications that support new business models, they discover new ways of doing things; they stumble into short cuts, undocumented features of tools and "cheats" that lessen the burden of developing new systems. This new knowledge will go unnoticed or will remain tacit in the team member's mind if not unearthed and made available to a wider audience. Software development is an art form, while both governed by the same operational guidelines, different team members will use different approaches in achieving a similar task.

  Of these two, one will be more efficient, consume less effort and will deliver better quality results – this is the best practice that needs to be identified and promoted. Organizations need

[66] Gottschalk P, 2005

to pay special attention to ground breaking projects as these have a high potential for generating new knowledge.

- **Reference material**

  IT people are not known for being good manual readers; their tendency is to "learn as they go along" and to find their way through the new technology being introduced. When people do read the manuals, they discover ways of working that provide new insight to existing problems. Enough reference material need to be provided to team members covering various angles of the software engineering discipline, team members should be encouraged to read to improve their skills. Some organization will have contracted time which technical resources are expected to use for self-development. There should be a management interest on how this time is utilized and what it yields in terms of performance improvement.

- **New tools**

  Introduction of new tools or upgrades of existing ones require that team members re-skill themselves in the use of such tools. As team members become more proficient in the use of these tools they accumulate know-how which is vital to the success of the organization. New tools bring additional features, simplify tasks and introduce new concepts which are meant to speed up delivery, they come loaded with embedded knowledge from the tool vendor, learning them equates to internalizing this knowledge. Organizations need to take stock of the growth of competency in newly adopted tools.

- **Training**

  Training is a standard way of increasing team knowledge on a particular subject. During training both tacit knowledge from the trainer and explicit knowledge from the accompanying training material is transferred to the trainee through a process of internalization, the result being that the trainee gains more knowledge on the topic concerned and therefore becomes more valuable to the organization. Training is an investment and the return is a more knowledgeable, more productive resource. The knowledge gained through this training is an organizational asset that needs to be accounted for.

- **External expertise**

  Organizations regularly engage external experts either as consultants or as outsource vendors to assist in the delivery of software systems. These external experts are on the main solicited because of their knowledge which does not exist internally. The client organization, through its team members that interact and work with the experts acquire new knowledge which could give it the edge over competition. Team members also gain additional knowledge through subscribing to external bodies such as news groups and special interest groups. This interaction with the external world increases the organization's pool of available knowledge. Processes need to be in place to formalize and guide the discovery of knowledge in the above high potential areas.

4.2.2 Knowledge capturing

Knowledge generated is useless if it is not captured and stored for reuse. Empirical evidence has shown that while organizations will generate knowledge and learn from their experiences, they also forget and lose these experiences hence the capturing and storing of knowledge into reference-able and reusable format is a critical component of an organization's knowledge management strategy[67]. One important hurdle that must be overcome by a knowledge management strategy is how to encourage people to willingly share their knowledge for the betterment of project and organizational performance[68]. Knowledge capturing can be facilitated through:-

- **Incentives.**

  Everyone has come to realise that knowledge is power therefore having knowledge enhances the team member's status hence the potential reluctance to share.
  Part of the knowledge management strategy of the software engineering organization should be to incentivise team members to volunteer their knowledge for the greater good of the organization. Incentives could be monetary or symbolic depending on which one creates the most stimulus. In many instances the value of captured knowledge far surpasses the cost of the incentive. A reward gives an indication of what behaviour the organization values and

[67] Alavi M & Leidner DE, 2001
[68] Kazi A, 2005

this in turn will communicate the value the organization puts on knowledge and incentivise team members to share more of the knowledge they have acquired.

- **Knowledge harvesting structures (Project review and close out sessions)**

  A lot of new knowledge gets generated by team members during the course of project delivery, the software engineering organization needs to have structures in place to query and capture this knowledge. Structures such as project reviews and project close out sessions are meant to harvest project knowledge and document learnings for review and reuse. Organizations also have a responsibility to foster a culture of knowledge sharing.

- **Directories of knowledgeable subject matter expert**

  Tacit knowledge is not easy to document, a strategy to counter this is to establish a directory of who knows what within the software engineering organization, this helps with knowledge mapping ensuring that the knowledge requestor can easily locate the provider. Considering that IT knowledge is ever changing, codified knowledge might soon be obsolete; linking knowledge to a domain expert ensures that the knowledge available to the organization is always current. The expert could either be an internal resource or external.

- **Knowledge repository**

  Explicit knowledge is about documents contain findings, reviews, short cuts, standards, guidelines, minutes, how-tos, best practices, references, templates etc. Once captured this knowledge needs to be stored in a sharable central repository setup to be easily accessible to the organization. A repository provides technical security to the knowledge assets and ensures that knowledge can be manipulated in ways that will bring value to the organization. The repository also becomes the single source of the truth for all the knowledge requirements of the organization.

- **Organization structure - Knowledge Manager**

  The software engineering organization is a knowledge intensive organization; for it to enjoy priority, knowledge management needs to be located at the right level of the organization. Considering also that knowledge will not necessarily be shared voluntarily, the organization needs to establish a role within the organization's management structure whose purpose it is to champion the cause of knowledge management within the software engineering

organization. Establishing the role of a Knowledge Manager ensures that knowledge management receives the right focus and that all knowledge management activities are properly coordinated. The role also becomes the interface for knowledge collaboration with other strategic partners namely business interest groups, academia etc.

- **Reviews and Updates**

  Knowledge gives the organization the edge, IT knowledge because of the fast pace of change in technology is very volatile and therefore to maintain the urge needs to be reviewed and revised and remoulded to the changing environment. Software engineering processes are influenced by the technology that implements them and when this technology changes, the knowledge embedded in the processes need to be revised and realigned.

  Incorrect information is dangerous, it can lead to faulty decisions, for this reason it is crucial that knowledge held in the organization's repository is reviewed and refined regularly.

- **Surveys**

  The usefulness of knowledge lies in its application, effort is wasted if the organization's knowledge does not get utilized. Part of an organization's knowledge management strategy is to ensure that the knowledge held in its repository remains relevant to the needs of the organization, this can be done through checking the frequency of use or surveying the user community. The purpose of these assessments is to review the effectiveness of the knowledge management strategy and to create a feedback loop to enable continuous improvement. Content that does not add value is trimmed out and areas of high interest receive a higher focus.

4.2.3 Knowledge sharing

It is not so much that people have difficulty expressing and articulating what they know, but that they may not be conscious of what it is that they know[69]. It is evident from the statement by Haldin-Herrgard that organizations cannot leave the communication and distribution of knowledge to chance, part of the knowledge management strategy needs to be the diffusion of the right knowledge to the right audience at the right time. Culture is identified as the most

[69] Haldin-Herrgard T, 2000

significant barrier to sharing knowledge[70]; it is for this reason that organizations need to pay special attention to cultivating a culture supportive of knowledge sharing. While the mission and values statement might say the right things from a strategic point of view, software engineering organizations need to make sure that these values are practiced on the ground rather than being lip service so that the required culture is achieved.

Some of the strategies that software engineering organizations need to follow to ensure proper diffusion of knowledge generated from their projects include:-

- **Lessons learnt sessions**

  Software development projects are a source of knowledge for many of the team members, it is therefore important that this knowledge is identified, harvested and shared with the rest of the organization. Lessons learnt sessions are structured sessions that afford the project team members the opportunity to share their learnings with the aim of externalizing and codifying the knowledge generated. Not only good information should be shared, bad experiences as well to ensure that these become part of the learning cycle. Capturing and sharing of lessons learnt ensures that knowledge gained from a project benefits a wider audience than the few directly involved in the project and this in turn ensures maturity of the organization as a whole. There is a danger that clients of outsourced projects might not benefit fully from knowledge generated through lessons learnt sessions because of the distance separating the two organizations. It is therefore important that the outsource engagement model tying the two organizations together provide for the sharing of this knowledge.

- **Knowledge Bulletins**

  The dissemination of knowledge contained in lessons learnt reports should not wait until the completion of a project, during the lifecycle of the project; knowledge bulletins can be released anytime there is a need for written communication of some aspect of technical knowledge to the wider software engineering organization[71]. This ensures that knowledge flows timely through the lifecycle of the project and that bad experiences are not repeated by parallel projects. Knowledge bulletins also serve to keep stakeholders and other interested

[70] Ruggles, R, 1998

[71] Kazi A, 2005

parties informed about the knowledge discoveries made within a project therefore helping to strengthen the business case and justifying the necessity for the project.

- **Knowledge portals and Wikis**

  Knowledge portals are intranet based websites the purpose of which is to expose the repository of knowledge that the organization holds on various topics. The portal allows for topical searches using menus and keywords and for aggregation of extracted knowledge to the user's taste. It is easy for an IT organization to use information technology to share knowledge as there is less resistance from its users. Once knowledge is electronically codified it can be posted into a knowledge portal where it can be accessed by the whole organization at leisure. Wikis are personal websites hosted by individuals or internal interest groups around specific topics, the implementation of wikis to share knowledge allow team members a level of control and ownership and this will encourage them to take ownership and actively participate in the sharing of knowledge.

- **Communities of Practice**

  Communities of practice are organic and self-organized groups of individuals who communicate regularly to discuss issues of mutual interest[72]; they are bound together by shared expertise and passion for a joint enterprise[73]. Communities of practice are driven by staff members and therefore do not carry the perceived stigma of management authority. Team members who had participated in project delivery and contributed to lessons learnt can be encouraged to present their findings at the relevant communities of practice. To ensure relevance care needs to be taken that knowledge presented to the community of practice is interpreted and contextualised to fit the purpose of the community and that the tone of the presentation is sharing rather than instructing. Apart from using the forums to disseminate knowledge, software engineering organizations can leverage the team spirit and the power of Communities of Practice to solve complex software development related problems. Utilizing their strategic relationships with external vendors, organization can send their specialist to "sit in" and participate on their partner's Communities of Practices and tap into the wider experience the vendors have. Communities of Practice also enforce the culture of sharing and

[72] Lave J & Wenger E, 1991

[73] Wenger EC et al , 2002

participation and there diffuse the tendency by some to hoard knowledge. While Communities of Practice are self-driven, organizations need to nurture them and encourage participation so that they can evolve.

- **Evangelists**

  Microsoft uses the idea of evangelists, a specialist role in a particular area whose purpose is to visit clients sharing knowledge on product improvements, roadmaps and new ways of working from other clients. Software engineering organization can use this concept by creating roles which would conduct presentations and demonstrations on new discoveries and new approaches to doing things. The evangelist's role is to create awareness and excitement rather than running in-depth training courses, they carry a high-level message intended to get the community excited to go try it out.

- **In-house training**

  Software engineering cannot rely solely on individuals changing their ways simply by observing, listening to evangelists and reading knowledge bulletins. To ensure that learnings are embedded and become part of the new practices, employees need to be trained and guided on the new ways of working and on how to leverage the newly discovered knowledge. In house training is topic specific and is meant to impart new knowledge and to correct unwanted behaviour observed during project delivery.

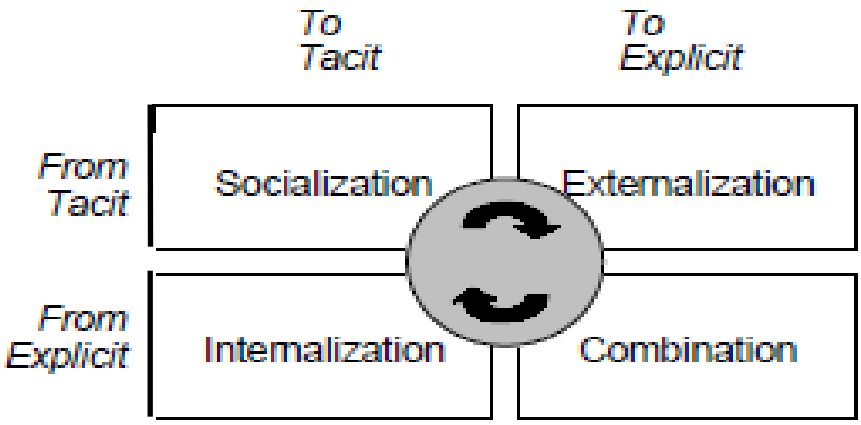


Figure 4.2 - The SECI knowledge creation process

(Source: Nonaka I & Takeuchi H, 1995)

In Figure 4.2 above, Nonaka and Takeuchi illustrate the creation of new knowledge through the combination of tacit and explicit forms of knowledge and the sharing of this knowledge through socialization and externalization. Socialization is the exchange of tacit knowledge through social interaction. Externalization is the conversion of tacit knowledge into artefacts through codification. Combination is the conversion of explicit knowledge into more complex forms of knowledge. Internalization is the conversion of explicit knowledge into tacit knowledge. The SECI model illustrates how organizational knowledge is constantly evolving into new forms through the interaction of its employees.

4.2.4 Knowledge application and utilization

The purpose of creating, capturing and sharing knowledge is to leverage off it for the greater good of the business. Knowledge management processes need to be creating value and advantage for the business otherwise the knowledge management effort will not be sustainable. Software engineering organizations need to constantly demonstrate value created by knowledge management strategies partly through:-

- **Review of ways of working**

  New knowledge modifies behaviour, as part of feedback from lessons learnt; organizations might need to review their process to take advantage of new ways of doing things. Processes could be simplified and thereby increasing delivery quality and reducing costs. Root cause for a nagging problem could be discovered and a permanent solution introduced. Abstraction could lead to software accelerators being developed and used as foundation in future software development initiatives reducing delivery times and minimizing complexities.

- **Benchmarking against best practice**

  Research organizations such as Gartner[74] regularly publish benchmarks of best performing software engineering organizations, these results could be used to measure the benefits of knowledge on performance and its contribution to the organizations financial results.

[74] www.gartner.com

- **Maturing the capability of the delivery organization**

  Knowledge sharing results in the improvement of the overall capability of the software engineering unit. The Software Engineering Institute's[75] Capability Maturity Model could be used to measure and quantify the process gains brought about by proper management of knowledge.

Knowledge management is not an IT initiative but rather an organization wide imperative. For an organization to succeed in effectively managing its knowledge, its knowledge management strategy needs to be linked to the enterprise strategy[76].

4.3 Knowledge Management Systems

Knowledge management assists an organization to effectively manage its organizational change and development; it also helps an organization to capture lessons learnt and to build an organizational memory from which it can drive growth and innovation. Organizational memory is a prerequisite for organizational learning, without it significant lessons learnt and other important insights would not be retrievable for future use[77]. Knowledge Management System does not refer to a single system but rather to a class of IT applications that support and enhance the organizational processes for discovery, storage, retrieval, sharing and utilization of corporate knowledge[78].

4.3.1 ICT in Knowledge Management

There are many compelling reasons why organizations use ICT as part of their knowledge management strategy; the CIO magazine survey indicates that a high percentage of organizations (67%) associate the investment in ICT for knowledge management with increased revenues and overall improvement in profitability[79]. Information and Communication Technology (ICT) has come to be recognized as a key enabler of knowledge management. The computer's ability to connect people across vast distances, store huge amounts of data and the speed with which it

[75] www.sei.edu
[76] Maier R, 2004
[77] Maier R, 2004
[78] Alavi M & Leidner DE, 2001
[79] CIO Magazine, 2001

moves this data between two points makes it a great enabler for knowledge management. IT has the ability to extend knowledge transfer channels beyond corporate borders allowing for collaboration that is not limited by time and space. More and more companies have instituted knowledge repositories supporting such diverse types of knowledge as best practices, lessons learnt product development and customer knowledge[80]. A Knowledge Management System can be defined as an ICT platform that combines and integrates functions for the contextualized handling of both explicit and tacit knowledge in the area that is targeted by the Knowledge Management initiative[81]. At the core, a Knowledge Management System needs to provide a dual purpose namely being the repository and reference for explicit organizational knowledge as well as being the expert locator for tacit knowledge[82], above all it also needs to be an efficient instrument for knowledge sharing.

4.3.2 Evolution of Knowledge Management Systems

Knowledge management systems (KMS) have evolved from being a single feature incorporated within an operational system to being standalone full feature platforms that integrate all the organization's knowledge requirements. The KMS industry has now developed to the extent that it has become difficult to understand similarities and differences between the different product offerings, new players are coming in all the time and the product arena is changing rapidly as new problems begin to surface[83]. The basic function of a Knowledge Management Systems is to capture explicit codified knowledge into a repository and provide means to search and categorise this knowledge into a meaningful format, the system also has to make tacit knowledge accessible by locating experts within the organization and linking these to people in need of the knowledge they possess. Following are operational systems that address specific features of knowledge management albeit not in an integrated fashion:-

- **Workflow management systems**

  Workflow management systems automate and coordinate related activities of a business process; the activities are triggered by an external event such as authorization of a payment

[80] Davenport TH & Prusack L, 1998
[81] Maier R, 2004
[82] Hansen MT et al, 1999
[83] Housel T & Bell AH, 2001

and are carried out by a person using resources such as documents, applications software and data[84]. Workflow management systems support well defined organizational processes and will use embedded intelligence to route work to its intended recipient and make decisions such as escalations based on programmed conditions. These systems also provide the participants with help and alternatives on the operation being performed.

Forms can be automated to have context sensitive help facilities allowing for an easy fill-in and intelligent routing.

Workflow Management Systems therefore externalize the knowledge of performing business functions and routing work which in many instances is tacit within the team performing that particular function. With a Workflow Management System an organization no longer has to solely depend on domain experts to understand the intricacies associated process flows and approvals.

- **Intranet**

  The most common objective for knowledge management initiative in many organizations is the capturing of organizational knowledge into a repository for later reference by a wider community[85].

  In this regard the intranet has come to signify an organizations internal repository of information; it is a source of knowledge where members of the organization can get information and assistance on how to go about performing their duties. The intranet provides access to the organizational databases allowing people to educate themselves on such things as company policies and to have access to online help on how to perform their duties. Organizations can use the intranet to post important notices and announcements allowing members in remote regional offices to be informed of what is going on at the head office. Various databases of information accessible through the intranet are composed of information that has been harvested from experts, making the intranet a tool for sharing valuable organizational knowledge. Wikis are an example of how the intranet can be tailored to provide contextualized information to specific communities within the organization allowing them to engage and share knowledge that is pertinent to their needs.

[84] Maier R, 2004

[85] Grover V & Davenport TH, 2001

Many of the organizational operational systems such as HR now have web interfaces which allow for access through the intranet thereby providing self-service to the users. This in turn takes away mundane tasks from the scarce and expensive resources allowing them to focus on more value adding roles. A portal front end can be deployed to aggregate all the underlying operational systems into a single web based view.

- **Groupware**

  Groupware is a category of software that supports collaboration between groups and teams. Systems that fall into this category include E-mail systems, Calendar sharing systems, Video conferencing and Message boards[86]. Groupware systems have evolved to incorporate instant messaging and online chatting facilitating collaboration amongst teams with common interest. Products such as Microsoft Exchange and IBM's Lotus Notes and Novell's Groupwise are the dominant leaders in this space.

- **Data warehousing**

  A data warehouse is a collection and refinement of data from various transaction processing operational systems into a single reference-able base. It allows the organization's decision makers to gain an integrated view of the different parts of business and to extract consolidated information on the areas of interest. Unlike the transaction systems from which it extracts its data, the Data Warehouse is optimized for high performance and decision making through integrated analysis of large quantities of corporate data.

- **Business Intelligence**

  Business Intelligence systems are a maturity of the Data Warehousing concept allowing organizations to analyse corporate data picking up trends and forming insights.
  Business Intelligence takes into consideration the organization's corporate goals and tries to establish trends that will assist in the advancement of such goals. Business Intelligence systems are not meant to replace the organization's experts but rather to augment their actions and empower them with the right information to make effective decisions.

- **Search Engines**

  Search engines allow users through simple queries to search a domain of web sites located either internal or external to the borders of the organization for information that they require.

[86] Watson R, 1999

The search results can be further refined until the user narrows the selection to the information required. To ensure that the search always yields the latest information, search engines constantly scan the web for new pages using spider and robot programs[87].
The search engine allows the users to help themselves by locating relevant information from database hosted in different databases within and outside of the organization. More intelligent search engines will analyze user queries and build a profile of the subjects the user is interested in thereby rendering a more relevant set of replies, even going as far as suggesting other relevant topics.

- **Computer Based Training**

  Computer based learning tools are technology applications that support teaching and learning offering online learning content in an interactive way. Organizations load the learning content into the Computer Based Training system allowing users to teach themselves at their convenience, this way the organization can up-skill its workforce with minimal disruption to its daily operations. Subject matter experts would normally be attached to hosted content allowing students to post questions and get clarification on issues requiring clarification. Students can take tests to gauge their level of understanding and the organization would access these tests to assess the level interest and the proficiency of the workforce in a particular subject. Computer Based Training plays a significant role in regulated industries such as banking where there is a requirement for staff to show proficiency and understanding of certain industry governing regulations.

[87] Brenner W et al, 1998

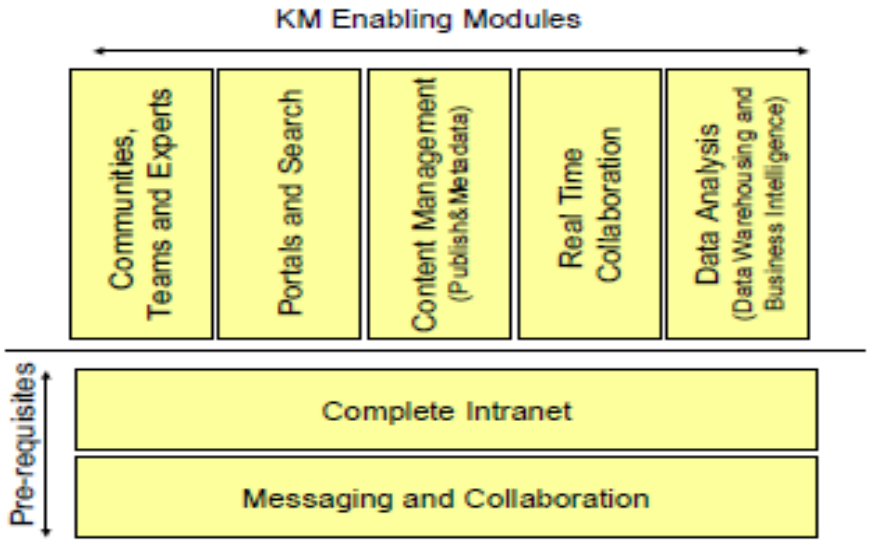


Figure 4.3 - Enabling modules for a functional Knowledge Management System

(Source: Brenner W et al, 1998)

As illustrated in Figure 4.3, the knowledge management modules discussed above combine yielding an integrated Knowledge Management System. The adoption of technology tools to aid knowledge management is a process that an organization matures into over a period of time.

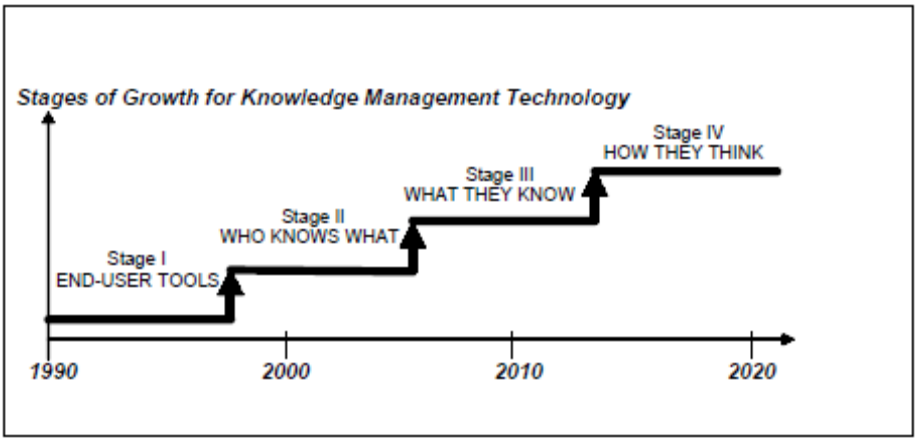


Figure 4.4 - Knowledge Management Technology stage model.

(Source: Gottschalk P, 2005)

The four stage model in Figure 4.4 illustrates how an organization starting at the simplest form of IT knowledge management services will evolve over time to offer a sophisticated knowledge management system. The maturity is organization dependent, some organizations will progress faster than others and some may decide to halt their progress at a particular stage as their needs become fulfilled. Progress is also influenced by industry maturity and the availability and stability of the technology used at the different stages. As an organization matures through the stages it becomes more reliant on technology to fulfil its knowledge management strategy and technology begins to play a bigger role in the fulfilment of an organization's knowledge management objectives. The stages can be analyzed as follows:-

Stage I

Stage I is about providing basic end user services that focus on personal productivity such as office automation tools allowing workers to create and share documents electronically.

Stage II

Stage II involves harvesting and codification of knowledge creating a database of available know-how, it is also about establishing a yellow-pages list of available expertise for easy location and referencing.

Stage III

Stage III is about implementing sophisticated tools to search and mine available data and to also expose it to external partners for cross organization collaboration. Workflow systems are also implemented to automate knowledge management support process.

Stage IV

Stage IV involves decision support through the implementation of expert systems and artificial intelligence. It is about scenario planning and analysis through rule based and case based reasoning systems.

4.3.3 The architecture of Knowledge Management Systems

Knowledge management systems have evolved from the disparate point solutions listed above into integrated systems that provide end to end management of an organization's knowledge assets.

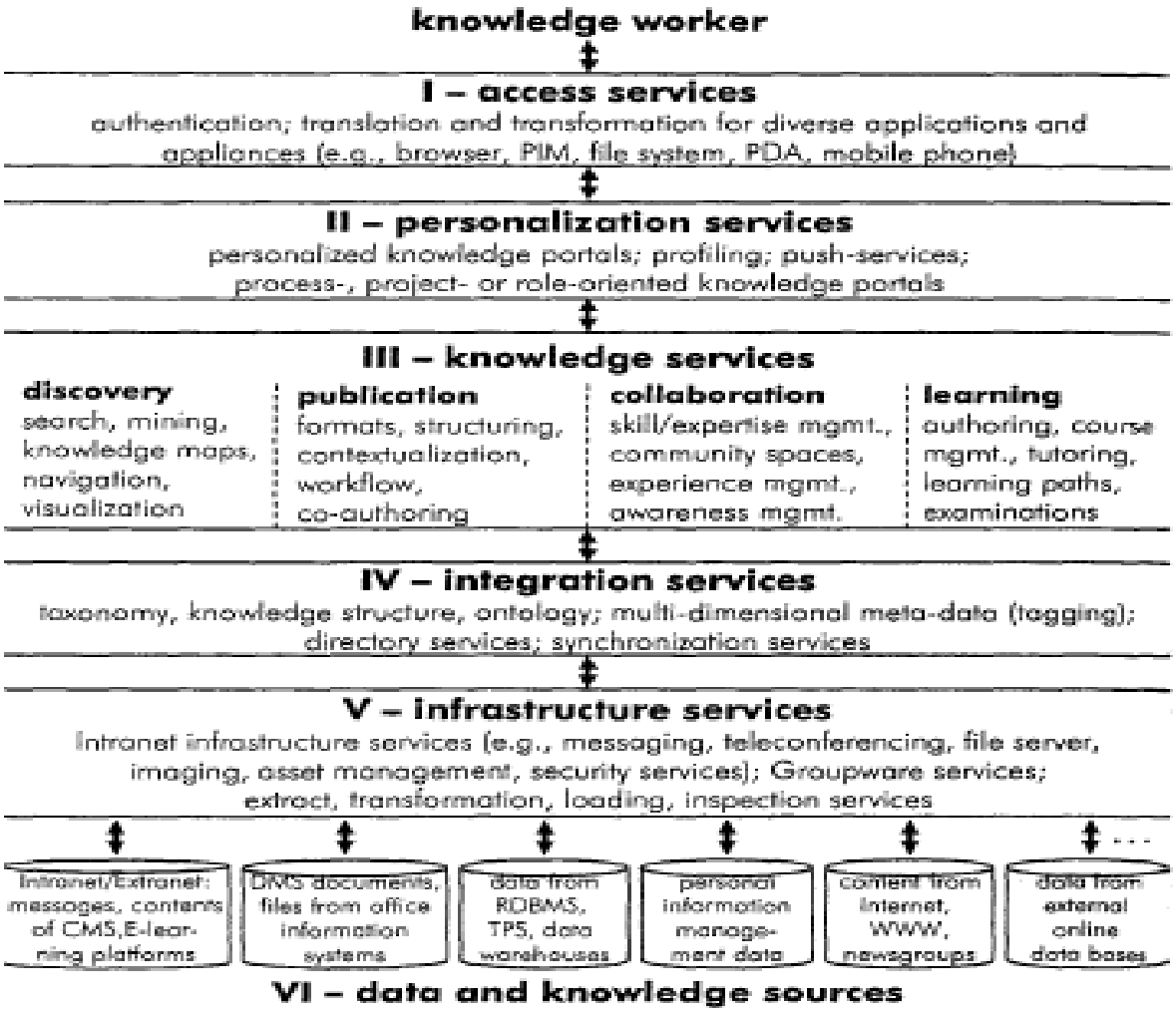


Figure 4.5- Architecture for centralized Knowledge Management Systems

(Source: Maier R, 2004)

Illustrated in Figure 4.5 is a typical architecture for a centralized Knowledge Management System, the six layers of the architecture illustrate the services that the Knowledge Management System provides and how these relate to each other.

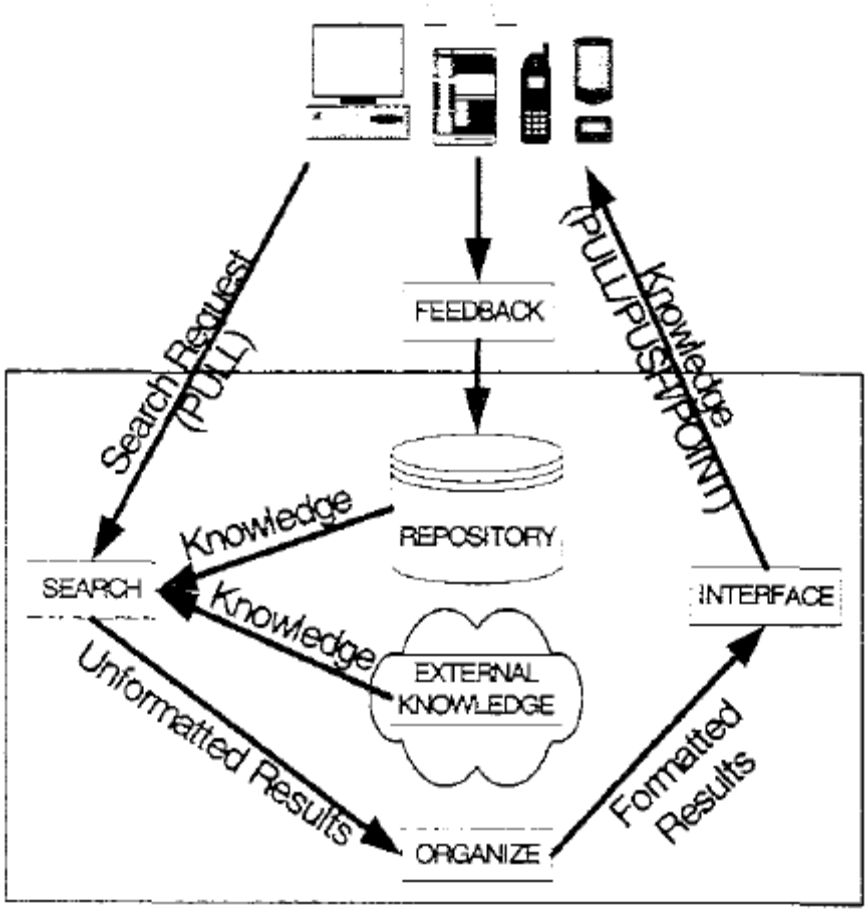


Figure 4.6 - Knowledge Management Systems information flows

(Source: Gray P & Tehrani S, 2003)

In Figure 4.6 above Gray and Tehrani shows a different view of the Knowledge Management System Architecture highlighting user interfaces and process flows involved in rendering the required knowledge. The user interface into the system is via the traditional desktop as well as the more modern forms of interfacing viz. cell phone, PDA and telephone for providing feedback on quality and suggestions for improvement. Knowledge is captured to a central repository and queries are submitted to the search engine which provides access to both the internal repository as well as external knowledge sources. The results are organized in order of relevance and formatted to the requirements of the interface (desktop, PDA) before being rendered. There is also a subscription facility which provides a push function to alert users of new knowledge that has been added in line with their preferences.

It is important to highlight that the knowledge management system is a support tool; it supports the organization's knowledge management processes rather than replace them.

4.3.4 Knowledge Discovery Systems

Organizations generate large volumes of data from various sources including its transactional and administrative systems. Inside this data lies valuable information that can be turned into critical knowledge which can be used to improve processes and to help drive business strategy. Knowledge Discovery Systems delve into these large corporate databases; processing and analysing available data to extract nontrivial, implicit, previously unknown and potentially useful knowledge in order to support crucial business decisions[88]. Knowledge Discovery Systems mine available data using complex algorithms and patterns to discover knowledge.

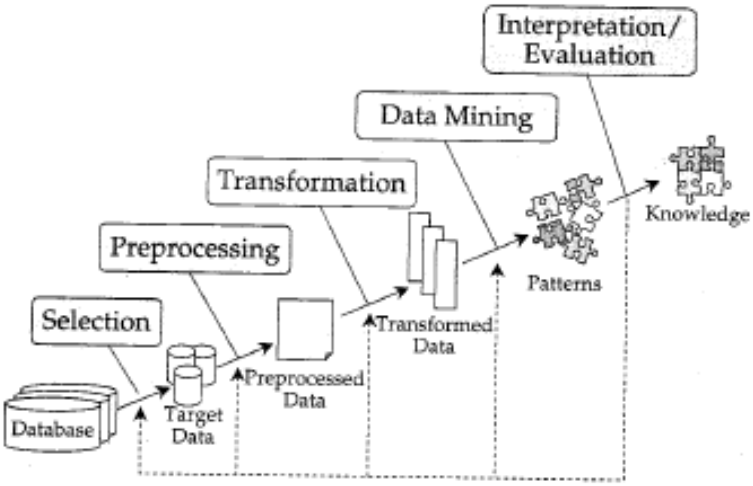


Figure 4.7 - Process of knowledge discovery

(Source: Wei C & Piramuthu S, 2003)

Figure 4.7 above shows the process followed by the Knowledge Discovery Systems in discovering of new knowledge from existing information.

Selection

The Selection phase extracts data from several target sources on which new knowledge is to be mined using a set of predetermined criteria.

[88] Chen MS et al, 1996

Pre-processing

The pre-processing phase integrates the extracted data and arranges it into a common format. The data is also scrubbed to remove all attributes that don't match the criteria.

Transformation

The Transformation phase transforms and rearranges the pre-processed data into a format that complies with the data mining techniques to be used.

Data Mining

Data mining scans the pre-processed data extracting patterns based on a adopted techniques.

Evaluation and Interpretation

The evaluation phase interprets the mined patterns and checks them against existing knowledge for consistency and to establish potential conflicts and finally present these as part of new knowledge.

- **Establishing relevance to Software Development**

  Knowledge discovery is regarded more as a business tool rather than a technology solution. Areas whose activities are cyclic in nature such as Finance, Marketing and Healthcare stand to benefit more from using Knowledge Discovery Systems as a means of harvesting new knowledge. While Knowledge Discovery Systems can be used in Software Development environment the benefits however will be minimal as IT Projects tend to be unique with a limited lifespan. Mining for patterns from previous initiatives is likely to yield little value, predictions can be made for instance about the failure rate of some software application but in the bigger scheme of things, the business case for implementing Knowledge Discovery Systems carries a bigger weight if motivated as part of an organization's overall management strategy. Non automated means such as brain storming sessions and innovation groups would however be good forums for achieving the same goals of knowledge discovery.

4.3.5 Knowledge Capturing Systems

Knowledge Capturing Systems aim to extract and preserve both explicit and tacit knowledge so that it can be secured and made available to a wider community. With reference to the SECI model described earlier, the Knowledge Capturing Systems facilitates the process of

externalising and internalising knowledge from various sources including employees, consultants, competitors, customers and suppliers. The knowledge is externalised into artefacts and once captured is internalised by internal resources into tacit knowledge as part of their learning process. Below are some of the Knowledge Capturing Systems that can be used in Software Development environment:-

- **Knowledge repositories**

  Knowledge repositories are specialized electronic storage mediums where explicit knowledge of various formats both structured and unstructured is stored. The repository provides multiple accesses via different forms of interfaces linking knowledge artefacts in a manner that allows for cross association. Knowledge repositories are hosted in fast access database management systems and indexed for easy and efficient access to the hosted knowledge content. The repository can be located at a central point or it can be architected as a distributed facility serving different locations with content contextualized for the various locations.

- **Mind Maps**

  A Mind map is a hierarchical representation of concepts. The concepts represented as nodes on the map are joined together via lines, these lines represent associations and relationships that exist between the concepts. Mind maps help in the visualization of complex concepts and representing these in simplified understandable formats. Figure 4.8 depict a visualization of core IT operations while Figure 4.9 illustrate software development lifecycle stages.

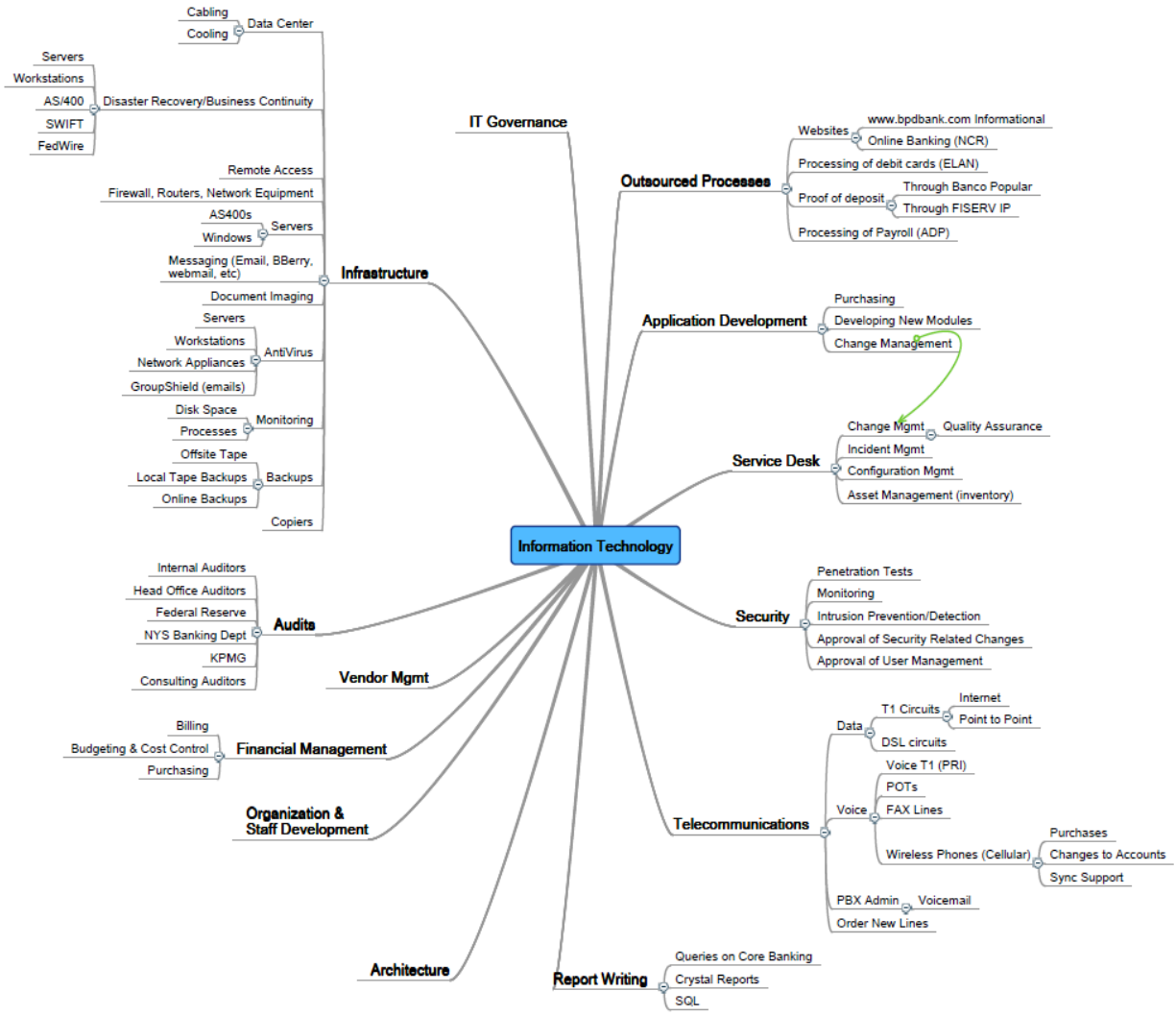


Figure 4.8 - Mindmap representation of IT operations

(Source: Constructed by author using MindMap tool)

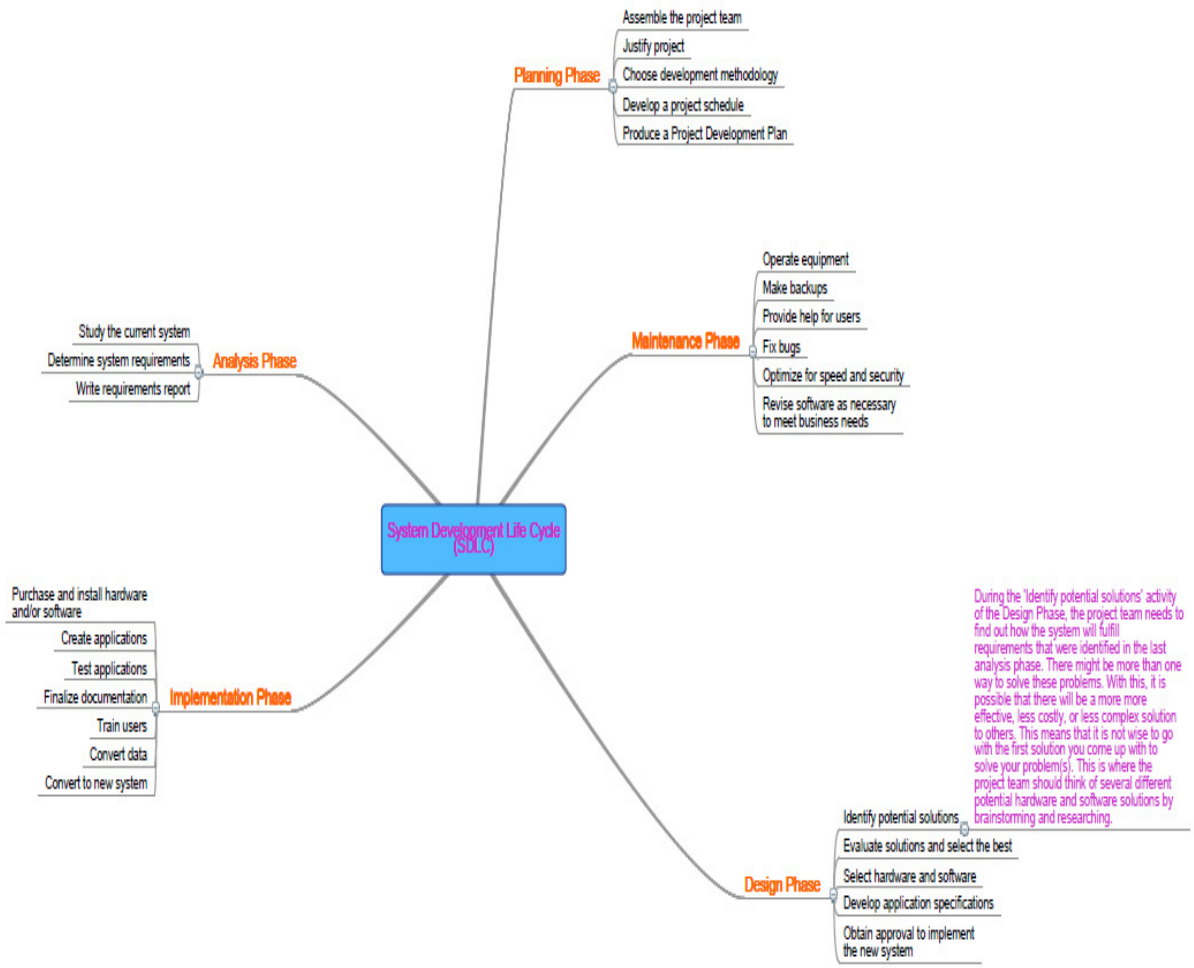


Figure 4.9 - Mindmap representations of the Software Development Lifecycle

(Source: Constructed by author using MindMap tool)

- **Story telling**

  Story telling is the oldest form of knowledge capturing; it dates back to early civilization where the elders used storytelling to pass on their values and wisdom from one generation to the next. The power of storytelling is that it captures both the knowledge content as well as its context[89]. Software Development project members usually have a lot of stories to tell concerning their experiences within the project, to accommodate these stories with potential knowledge content, the official project closure process could be complimented with a story

[89] Becerra-Fernandez I et al, 2004

telling session to ensure that the tacit knowledge that resides with the unsung heroes of the project is also captured. Special techniques need to be used in order to encourage the interviewees to volunteer as much of the knowledge content being captured as is possible. Once the narrative has been formatted and captured, it needs to be indexed so that it can be easily accessed by the relevant knowledge consumers.

4.3.6 Knowledge Sharing Systems

Software development is a collaborative exercise where each activity builds on the efforts of the previous one. Knowledge sharing is therefore crucial for the successful delivery of the project and the continued growth of the team members. Knowledge Sharing Systems are designed to help users share their knowledge, both tacit and explicit. They establish a market place where knowledge seekers and knowledge owners come to trade[90]. Knowledge sharing is one of the key pillars of a learning organization; the driving principle behind knowledge sharing systems is that the sharing must be easy, timely and relevant to the needs of the participants. In earlier chapters, differences were highlighted between information and knowledge; Knowledge Sharing Systems should therefore not dish out tons of information but should rather provide knowledge that has context and relevance. Below are some of the systems that facilitate the sharing of knowledge:-

- **Search Engines**

  Search engines have come about in response to the unprecedented increase in electronically available information caused in the main by interconnected databases and information repositories. In its basic form a search engine indexes every word in a repository and tries to match these to the query submitted by a user, sophisticated search engines use sophisticated algorithms to bring context into the search in order to increase precision and return more relevant results. Search engines allow knowledge stored in repositories located internally and externally to be shared by a wide audience, users can submit a query on a subject of choice and the search engine will mine the repository and present possible outcomes for further analysis.

[90] Dignum V, 2002

- **Content Management Systems**

  Content Management Systems are a special type of repository with the ability to categorise and link both structured and unstructured information. These systems are normally complemented by workflow and collaboration tools to aid in the sharing of the underlying information and knowledge. Software development teams can load all their Manuals, Standards, Software, How-to instructions, Training material and Lessons learnt into a single system and be sure that the knowledge will be securely accessible to all teams regardless of where they might be located. Newly discovered knowledge is also instantly available for leveraging by teams in other projects.

- **Portals**

  Knowledge Management Portals act as a single point of entry into the knowledge repositories of an organization. Portals can be broadly focused, in which case they cater for a wide variety of underlying knowledge or narrowly focussed in which case they cater for a niche set of needs. Portals save users from having to navigate through different sites to access the content they require, their ability to aggregate content across repositories and formats imply that users get the benefit of a consolidated view from a single search. Vendor repositories with hosting best practices, latest software patches and support documentation can also be made available through the portal giving internal resources access to expert knowledge from within the virtual boundaries of their organization. Content within a portal is protected by a role based multilevel security arrangement driven from a single sign-on architecture which means that users do not need to provide log-in credentials to each of the underlying repositories supported by the portal. This multilevel security also ensures that users only have access to the content they are authorised to.

- **Collaboration Systems**

  Collaboration systems facilitate interaction and sharing of knowledge between individuals and groups within and outside an organization. Collaboration systems allow for synchronous collaboration where the interaction between participants is real-time (e.g. Video conferencing, Instant Messaging, Electronic whiteboard) or asynchronous collaboration where messages are posted and stored for later review and reply (e.g. Email, Discussion threads, Document library).

With these tools a team member can post a question to an individual, internal group or the world and get back responses giving different perspectives of the answer.

- **Lessons Learnt Systems**

  A lesson learnt is knowledge or understanding gained by experience. The experience may be positive in the form of a success or negative in the form of a failure[91]. Organizations harvest lessons from their projects, verify these for correctness against a team of domain experts and store these with context for reference by a wider community. The benefit of lessons learnt is that achievements are celebrated and the teams use the knowledge shared as a foundation to create new knowledge. Celebration gives recognition to those who contributed to the success and reinforces the notion of sharing, encouraging a positive organizational culture that is conducive to knowledge creation; on the other hand failures are noted as pitfalls enabling mistakes to be learnt from but not to be repeated. In software development lessons learnt are the basis for new frameworks which allow for abstraction of complex concepts and reuse of existing knowledge. This contributes towards quality software, faster delivery times and cost effective solutions.

- **Expert Locator Systems**

  It has been stated before that not all of the organization's knowledge is codified and available as explicit knowledge that can be accessed from repositories; some of this knowledge resides as tacit knowledge with the various domain experts that are either part of the organization or its extended community of stakeholders. Expert Locator Systems catalogue available expertise within the organization and link these to those to areas where their assistance is required. The system can be used to match experts to available vacancies and to mitigate risk in areas where the expertise is limited to a single individual posing a risk to the organization should the individual decide to leave. Organizations need to adopt a standard knowledge taxonomy which they can use for classifying both the different categories of knowledge and the levels of expertise that are recognized within an organization, this assists the users in accessing the right knowledge and eliciting the right level of assistance.

[91] Becerra-Fernandez I et al, 2004

In Software development different technologies such COBOL, C++, Java and dot.Net are often used, each of these technologies have people that are regarded as experts by their peers and the industry, it is therefore important that such domain experts are identified and located so that they can be made available to assist in areas where their knowledge is required. When a project starts, it is important to ensure that the right quality of resources is allocated to ensure success. When outsourcing, Expert Locator Systems help to identify vendor expertise that would need to be paired with local resources to ensure a successful knowledge transfer, the criteria for successful delivery would include successful transfer of knowledge to the identified resources.

- **Communities of Practice**

  Communities of Practice are groups of people bound together by a common interest who come together to share experiences and learn from one another. Communities come together face to face or virtually and are driven by a need to share problems and insights[92]. Communities of practice foster team spirit and a sense of belonging which in turn encourages socialization with the benefit being a free flow of tacit knowledge amongst team members. The team's social networks help to grow the community and as such niche sub-communities who specialize in a particular field of interest are sometimes spurned. Because Communities of Practice are voluntary, they do not have the management stigma and are likely to attract all kinds of interested audiences and might help bring to the fore specialists from the fringes that might be "anti-establishment". Technologies such as WIKIs support Communities of Practice by providing a collaborative space that spurns across organizational boundaries. Software development environments would have a number of communities ranging from Project Managers, Analysts, Designers, Developers and Tester communities of practice. Each of these would carry a specific agenda and cater to a specialist audience even though there would normally be collaboration between the communities.

[92] McDermott R, 2000

4.3.7 Knowledge Application Systems

The value of knowledge lies in its application rather in the knowledge itself[93]. Knowledge Applications Systems support the process through which individuals utilize knowledge possessed by others without acquiring that knowledge[94]. Allowing more people to have access to and utilise available knowledge ensure optimum utilization of knowledge. By embedding its routines into an IT solution; organization are able to have lower level staff performing relatively complex tasks with the aid of knowledge application systems. In the context of software development, Integrated Development Environments (IDE) which are used by developers to write programs have embedded routines that guide developers on the best use of the programming language suggesting possible choices for function calls also highlighting and automatically correcting syntax violations. Helpdesk operators are able to support a software application resolving user issues through utilising a helpdesk system which gives pre-packaged answers to common problems, operators do not need to know the internals of the system but with the assistance of a Knowledge Application System are able to tap into the developer's know-how and use this to solve user issues. A process monitoring tool can highlight issues to an operator once a pre-programmed threshold is reached giving instructions of what should be done to resolve, the operator only has to have basic understanding to follow the instructions, he does not need to have intricate knowledge of the underlying machinery.

4.4 Conclusion

Software development projects generate new knowledge and organizations run a risk of losing this knowledge if they don't have processes in place to capture it, putting in place knowledge management systems ensure that this knowledge is captured with in a structured format. IT is an enabler of knowledge management, it has the capability to collect, structure, store and distribute knowledge efficiently to wherever the knowledge workers are located; software development organizations leverage this capability to implement a formal approach to managing knowledge. Applying Nonaka's SECI model we have seen how knowledge generation is an on-going process with knowledge constantly changing form through codification from tacit to explicit and being assimilated into tacit forms through internalization and also new forms of knowledge being

[93] Gottschalk P, 2005

[94] Becerra-Fernandez I et al, 2004

created through combination of existing knowledge. This constant morphing takes place all the time during software development and without Knowledge Management Systems organizations would not be able to capture process and apply such knowledge effectively. IT is utilized to support the key processes of knowledge management namely: - discovery; capturing; sharing and application of knowledge. Software development is an IT activity; it makes sense therefore for it to be setting the pace on utilising ICT to support its knowledge management efforts. There are IT solutions that cater specifically for each of these key processes; however an effective knowledge management system needs to be one that provides an integrated environment covering all the key knowledge management processes as highlighted above and providing easy access to knowledge workers wherever they might be located in the process of doing their job, with this KMS support in place software development organizations can focus on their core business of developing systems knowing that all experiences are captured and leveraged. The architecture for knowledge management system should be such that it facilitates access through traditional means as well as the new interfaces such as PDAs. It also needs to have the search capabilities to access both local as well as external knowledge repositories with a feedback loop for on-going improvements.

There is a growing realization, as indicated by the CIO magazine survey, that knowledge management systems are not only about the management of knowledge within an organization, their contribution impacts revenue growth and the overall profitability of an organization. There are however sceptics who question the supposed role of ICT in furthering knowledge management, those who feel that the proliferation of IT based knowledge management systems will result in large quantities of useless information[95], the emphasis being that organizations need to implement thorough screening mechanisms to ensure that the content that goes into their knowledge repositories is of high quality. The scarcity of skilled resources has generated a tug-of-war amongst organization to attract the best brains. Knowledge application systems allow organizations to make best use of these limited resources by packaging their knowledge and providing it as an automated solution to the specific organizational areas. Part of the reasons why organizations outsource software development is to gain access to expertise, with knowledge management systems they are able to capture this knowledge and use it to build internal capacity.

[95] McDermott R, 1999

It has been indicated in this chapter that software development is a collaborative exercise and that for knowledge sharing to take place, there has to be externalization or socialization of that knowledge, organizational culture also plays an important role in facilitating knowledge sharing. In the next chapter we look at how outsourcing software development to a different geographical location affects knowledge sharing between the teams and how knowledge transfer can still be maintained despite the teams operating across geographically dispersed locations.

Chapter 5
Sharing knowledge between onshore and offshore teams

5.1 Introduction

The previous chapters have demonstrated how the process of software development requires knowledge as input and how it generates knowledge through the collaborative efforts of the various experts involved. It has also been shown how vital it is for organizations to retain this knowledge for future leveraging and the ICT tools that can be deployed to aid in the retention and leveraging of such knowledge. This chapter considers the process of outsourcing software development to a different geographical location and the impact of such a strategy on the organization's knowledge assets. The chapter also evaluates the changes that outsourcing necessitates on the organizational structure of a software development business unit, the cultural impact this has on the organization and the effects on the organization's knowledge flows. Outsourcing is one of the strategies at management's disposal which can be used to acquire software development services from a vendor instead of utilizing internal resources, it is a build versus buy decision that organizations go through as part of evaluating the best option to follow. Outsourcing opens new talent sourcing options for organizations where they can have access to the most qualified resources from across the world at the most cost effective rates. When a decision is made to outsource software development, organizations need to evaluate the adequacy of its knowledge management strategy in view of this decision and identify areas where the strategy might need to be reviewed. This chapter identifies areas where software development outsourcing is likely to have the most impact and puts forward strategic interventions that an organization can implement to facilitate free flow of information between the dispersed project team members and to ensure effective sharing of knowledge while guaranteeing the security of the knowledge assets.

5.2 Global software sourcing models

Outsourcing has become one of the most important management ideas and practices of the past 75 years[96]; it features strongly in every manager's list of strategic options. With the abundance of IT skills and their cost arbitrage, places like India and China have become choice locations for organizations to consider outsourcing their software development projects to. Outsourcing allows organizations to achieve high quality results at a lower cost while also reducing headcount[97] thus making the organization to be more competitive under extreme economic conditions.

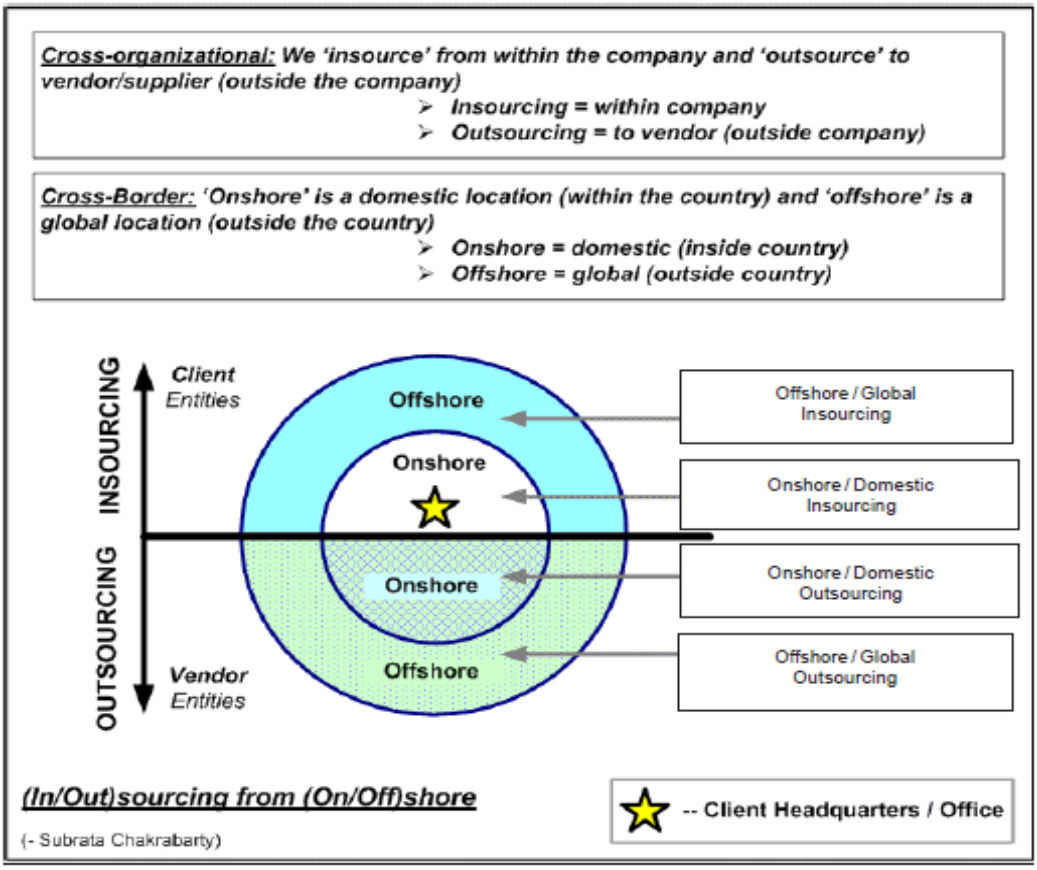


Figure 5.1 - Global software sourcing model

(Source: Kehal HS & Singh VP, 2006)

[96] Sibbet D, 1997

[97] Eltschinger C, 2007

As highlighted in Figure 5.1 above, outsourcing can take several forms with organizations using different permutations to access offshore software development resources.

- **Outsourcing versus Insourcing**

  When outsourcing; IT work that is traditionally done internally to the organization is moved to an external service provider. When insourcing; IT work is brought internally to be performed by the organization's IT department or subsidiary. Outsourcing and Insourcing can be further categorized into offshore and onshore depending on the geographic location of the service provider.

- **Offshore versus Onshore sourcing**

  Onshore Outsourcing implies outsourcing to a vendor that is located within the same geographic region as the client. Offshore Outsourcing implies outsourcing to a vendor located outside of the client's geographic borders, inclusive of overseas locations. Onshore Insourcing implies that both the organizations and its department or business unit managing IT are located within the same geographic region. Offshore Insourcing implies that the organization and its IT department or IT business unit are located in different geographic regions. When offshoring organizations can choose between four main options[98], namely:-

  - **Offshore partnership**
    Where the focus is on using an external partner to deliver specific projects or providing additional capacity on a longer term basis.

  - **Captive**
    Where an organization establishes its own IT subsidiary with the intension of insourcing work to offshore.

  - **Acquisition**
    Where an organization acquires an offshore IT entity with the intension of insourcing offshore.

[98] Eltschinger C, 2007

- **Build Operate and Transfer**
  Where an offshore partner is asked to build an IT capability with the intention of later transferring it back internally into the organization.

As can be seen above, outsourcing presents to the business a myriad of options which make the management of knowledge across the organizations involved difficult. Each one of the above options carries its own risks and as such potential benefits have to be weighed up against such risks. Outsource is intended as a partnership that is based on agreed deliverables, it is a deal that is entered into with the potential of being a long term engagement, it is for this reason that commitment is required from both parties involved and that the relationship needs to be mutually beneficial to both.

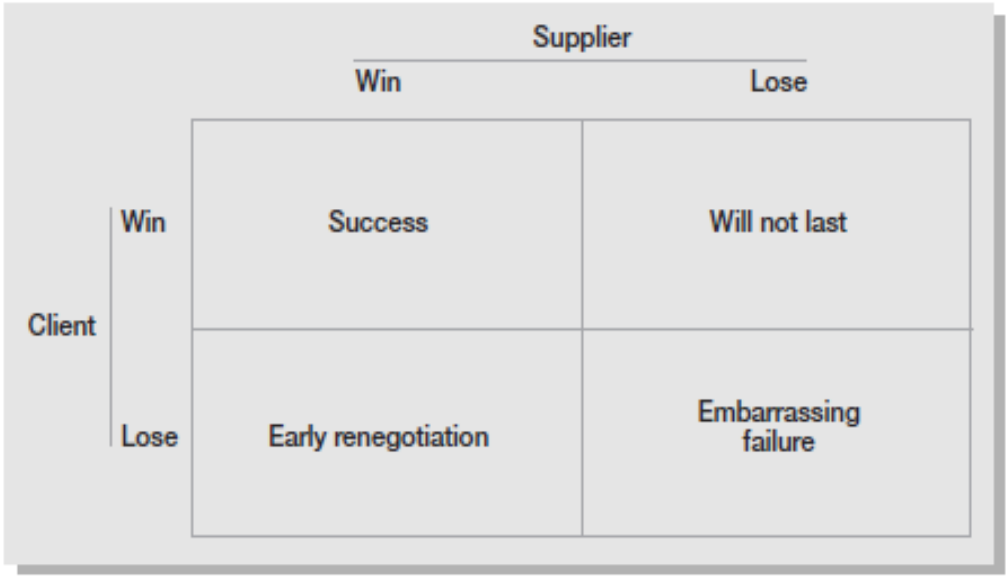


Figure 5.2 - Possible outcomes in an outsource relationship
(Source: Bravard J & Morgan R, 2006)

Figure 5.2 Shows the matrix of outsourcing outcomes to an outsource relationship and the potential consequences if a win-win arrangement is not achieved. For the relationship to succeed, both parties need to derive equitable value from it. The win-lose approach, while it might be appealing as a means of extracting maximum value from the deal is not sustainable and immediately causes souring of relations and a protracted renegotiation of the contract if the client

is on the receiving end. On the other hand if the vendor is on the receiving end; he will gradually withdraw and will hold back as much as possible in an attempt to recoup some of the losses, ultimately the deal will fail causing a lot of embarrassment and putting both organizations at risk. Failed deals have a massive impact on organizational knowledge as the separation is usually acrimonious without a proper exit mechanism even if forced by a contract.

5.3 Sharing knowledge across borders

Globalization has made people and countries to be interdependent, with the collapse of the Soviet Union and the proliferation of Information and Communication Technologies the world has become one. Globalization on the other hand has brought about an organizational evolution that has given rise to the phenomenon of a *networked enterprise*[99] characterised by a shift from mass production to flexible production driven by autonomous yet interconnected systems where knowledge is central to the operations of the networked entities and technology enables global collaboration between the elements of the system.

Knowledge sharing is a fundamental component of a network enterprise, there are many reasons why organizations and countries would collaborate and share knowledge, some of which include growing their economies, improving health, making new discoveries, reducing unemployment and expanding trade relations. In an effort to lower costs, an increasing number of organizations are adopting offshore software development as a part of their software delivery strategy[100] taking advantage of the abundance of cheap skilled labour from developing countries such as India and China. This has resulted in the software development effort being split between teams located in different countries and as such has created a need for the development teams to collaborate across borders. The effect has been that while this offshoring arrangement allows organizations to tap into an otherwise unavailable pool of skilled resources; it creates specific challenges with regard to the management of knowledge that is provided to and generated by the software delivery project straddled across the participating countries.

In an outsourcing setting, knowledge which existed internal within an organization's boundaries is moved to and exchanged with external organizations[101]. As demonstrated in previous chapters,

[99] Castells M, 2000
[100] Sahay S et al, 2003
[101] Bahli B & Rivard S, 2005

the software development lifecycle is a knowledge production line and therefore adoption of an offshoring strategy implies that the knowledge creation processes would be dispersed between the geographic locations involved in the outsource deal and as such would require specific knowledge management strategies. Software development builds applications which are strategic to the success of an organization as such the knowledge that is exchanged between the outsourcing organization (client) and the vendor is of a strategic nature and requires special measures of protection.

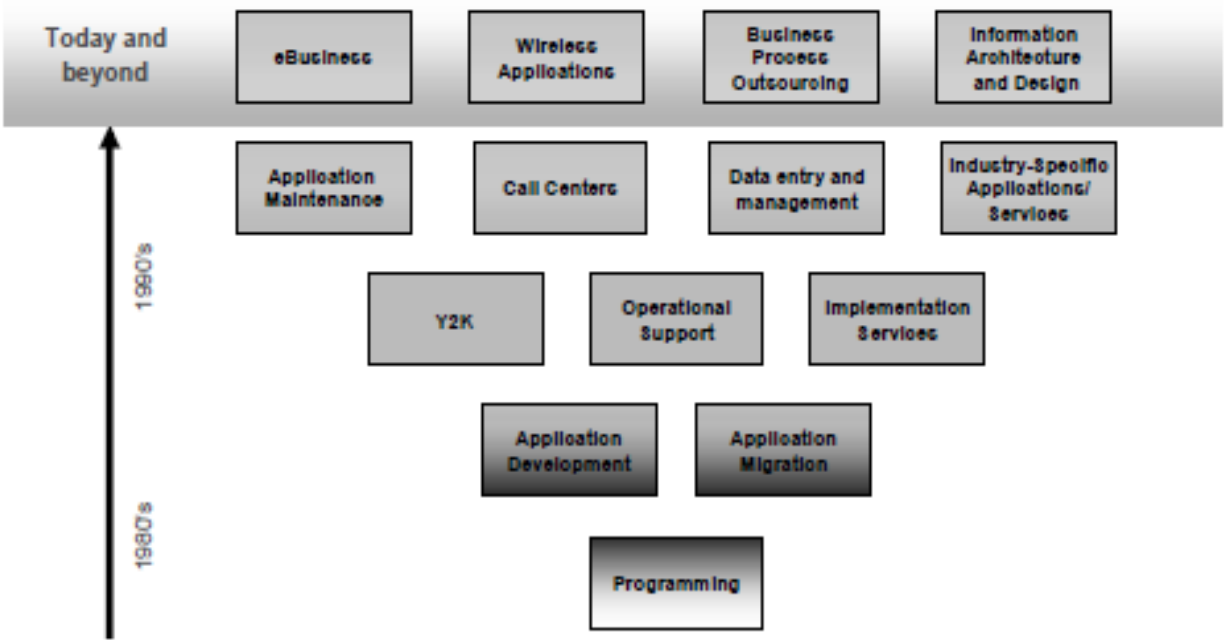


Figure 5.3 - Evolution of software outsourcing services

(Source: Vagadia B, 2007)

As illustrated in Figure 5.3, there has been an evolution of outsourcing services from rudimentary programming services towards knowledge intensive, business critical services. What is clearly evident is that while outsourcing in the early years was characterised by simple low value operations, it has since matured into a sophisticated engagement in which high value activities are being performed by third parties offshore, the implication is that the knowledge of

an outsourcing organization is no longer found only within the confines of its physical borders but instead it is spread across several geographic locations.

5.4 Knowledge flows within an outsource relationship

In an outsource relationship the client's aim is to leverage off the vendor's superior knowledge and management processes, however as the relationship matures; the benefits become mutual and both parties begin to tap into each other's competencies.

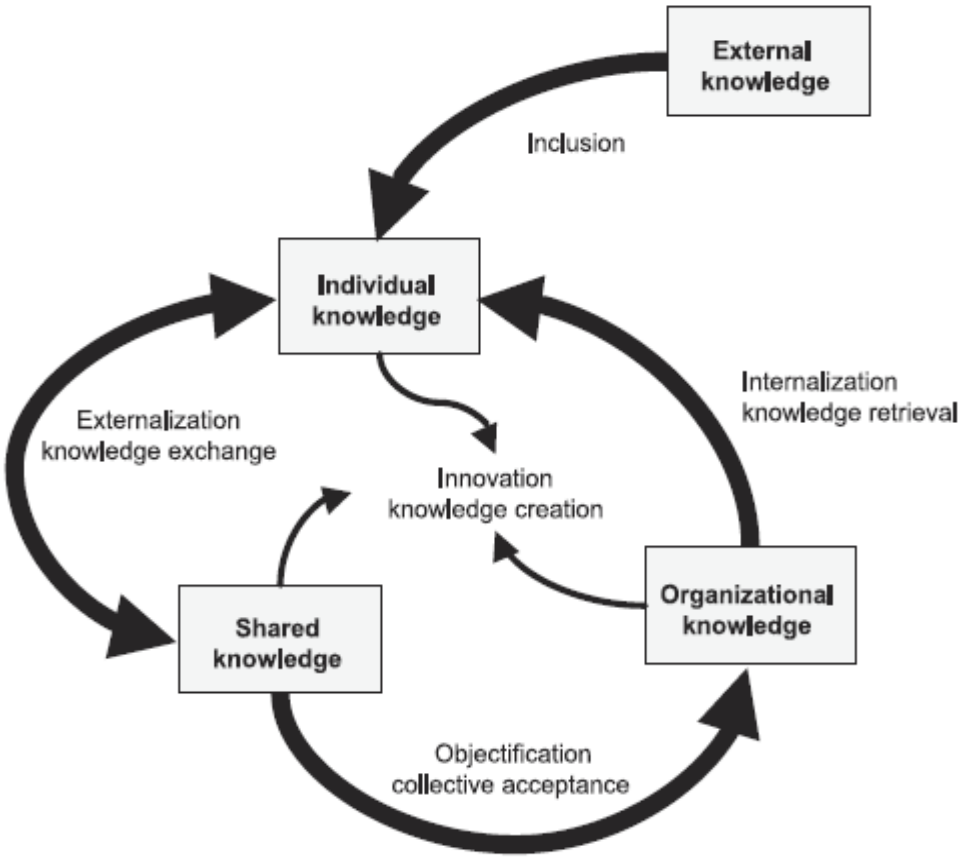


Figure 5.4 - Organizational Learning and Knowledge Sharing flows.
(Source: Huysman M & De Wit D, 2003)

Highlighted in Figure 5.4 is the constant flow of knowledge between an organization, its stakeholders and the environment, knowledge is constantly being gathered, exchanged, processed, recycled and utilized within the ecosystem. This constant flow of knowledge should

however not be confused with the information flows as described by Max Boisot[102] which refers to the continual adaptation of information as it evolves towards dominant design. There is dispute amongst knowledge management experts regarding what constitutes knowledge and information flows. Expanding on Fig. 5.4, one can observe that knowledge exchange between an organization and its outsourcing partner occurs at different levels. Following are some of the areas where knowledge transfer takes place between the organizations involved[103] :-

- **Knowledge transfer between individuals**

  This is knowledge that is shared between team members within the organization. This knowledge is created and shared as the team members continue to work together in various projects and share their experiences amongst each other. This knowledge is reinforced through such activities as team building sessions, company and departmental induction programmes, job rotation and apprentice schemes.

- **Knowledge transfer from individual to external parties**

  This is knowledge that is shared by individual members of the technical team with their counterparts within the vendor organization. This sharing allows the outsource vendor to get an insider's view of how the organization operates, who the key opinion makers and influencers are and how to avoid red tape in order to get things moving faster.

- **Knowledge transfer from external parties to individual**

  This is knowledge that is shared by the outsource vendor with individual employees internally. The experience of the outsource vendor gained from being involved in multiple projects of varying complexity empowers internal resources exposing them to new technologies and new methodologies. This knowledge sharing helps in improving the capability of internal resources and therefore the competency of the organization as a whole.

- **Knowledge transfer from the individual into the organization**

  This is the externalization of the individual's tacit knowledge into organizational knowledge artefacts such as manuals, repositories, tools and templates. This conversion ensures that knowledge becomes durable and sharable to the wider community.

---

[102] Boisot M, 1999

[103] Gottschalk P & Solli-Saether H, 2007

- **Knowledge transfer from the organization to the individual**

  This knowledge transfer makes available the knowledge held by the organization to the individual in order to empower him to deliver on the organization's mission. Once knowledge is externalized as proposed above, it becomes easier for the organization to share that knowledge with both internal and external partners to ensure competency building.

- **Knowledge transfer from the outsource vendor to the organization**

  This is knowledge that is formally transferred by the outsource vendor to its client. This knowledge includes improving process documentation, providing enabling tools and building new applications. When an outsource project is completed, it leaves the organization with an arsenal of new competencies through which it is able to tackle new challenges and conquer new markets. The vendor's knowledge artefacts such as templates and methodologies are transferred to the client organization thereby enriching its knowledge repository.

- **Knowledge transfer from the outsourcing organization to the vendor**

  The vendor builds its capabilities through the assignment it undertakes for its clients. From interacting with its client, the vendor gets to understand the business processes of the client, the business rules, industry trends and drivers, with this knowledge the vendor updates its base and improve its solution offering to the industry. The vendor also builds reusable software components which help them to build better quality products faster thereby fast-tracking future projects and increasing their standing in the market.

- **Knowledge utilization within the organization**

  With the newly acquired knowledge an organization is able to invest in itself and revamp its processes. The acquired knowledge is used to upgrade and realign organizational structures and as such allow the organization to compete proactively in the market.

As can be seen from above, the flow of knowledge in an outsource relationship is bidirectional, it involves and affects a wide variety of stakeholders and influences decisions across the spectrum of participating organizations. The people, processes and products from both organizations are greatly enhanced by the knowledge sharing that emanates from the outsource relationship. The

loss of information and knowledge can be detrimental for both parties unless specific strategies are put in place to guarantee the security of the knowledge pathways and to ensure sustainability of the knowledge management interventions implemented. As indicated earlier, software development is a collaborative exercise between the business community, the project team and the wider IT community within an organization. A key question worth probing is whether this collaboration is affected in any way by an organization's decision to outsource?

Activity	Outsourced	In-house
Project Management	30%	90%
Requirements Gathering	**17%**	89%
Architecture	19%	88%
Research and Development	25%	78%
Business Integration	16%	76%
Design	51%	77%
Systems Integration	35%	76%
Deployment	26%	75%
Testing	74%	71%
Modeling	26%	69%
Maintenance	53%	65%
Code migration	54%	42%
Coding	**94%**	41%
Internationalization	39%	34%

Table 5.1 - Percentage split between Outsourced and Inhouse activities.

(Source: Meyer B & Joseph M, 2007)

Table 5.1 illustrates how software development lifecycle activities in an outsource relationship can be distributed between the client and the outsource vendor in line with governance, economies of scale and cost considerations emphasizing the need for collaboration and knowledge sharing regardless of the software delivery model pursued by the organization. It is clear that there is a shared responsibility across all the lifecycle activities necessitating a need to manage information and knowledge strategically at all levels of the software development lifecycle.

5.5 Categorising knowledge to better enable sharing

Critical to the success of the outsourced software development project is the successful transfer of knowledge from the internal team of IT and business domain experts to the offshore team undertaking the software development effort. The knowledge to be transferred is both externalised and tacit and both forms require understanding and internalization by the delivery team within a short space of time for the project to be a success. While the vendor organization undertaking the outsource function will have in-depth knowledge about the industry the client is operating in and the tools to be used, it is critical that he understands the specific context in which this knowledge is applied within the client organization. The facilitated knowledge sharing sessions at the beginning of a project attempt to fast track the transfer of knowledge so as to quickly enable the delivery teams and ensure success of the project. In order to deliver successfully, the chosen outsource vendor needs to be empowered with the right knowledge so that he can fulfil all the project requirements as set out by the client. Categorizing knowledge ensures that all the knowledge elements critical to the success of the project are covered and accounted for. The following key categories of knowledge are shared as part of vendor enablement:-

- **Business and Systems requirements knowledge**

  The primary deliverable of a software development process is a set of requirements contained in the various documents prepared as part of the systems analysis stage of the software development lifecycle. The knowledge contained in this documentation and communicated to the vendor through arranged sessions gives the vendor the rationale behind the exercise, the problem areas to be addressed and the ideal solution required by the client business. From this, the vendor has a clear understanding of the client's requirements and the reason for the investment.

- **Infrastructure and Applications landscape knowledge**

  This is the knowledge regarding the IT infrastructure required to run the system as well as the infrastructure and applications currently deployed by the client that the system being developed will be interfacing and interacting with. This knowledge on the co-existence of these systems is important for the long term success of the new system. This eco-system

needs to seamlessly integrate and work together in delivering the client's current and future needs. The vendor will combine the knowledge shared by the client with his experience from previous similar engagements. This knowledge is also crucial in determining whether the option of supporting the solution offshore will be a viable one.

- **Programming and Tools knowledge**

  The vendor needs to have a thorough understanding of the development tools to be used in delivering the project so as to deliver an optimal solution at minimal risk to the project. The client also looks up to the vendor to empower his staff on the best ways of utilizing the chosen toolset. The vendor needs to know about any standards that the client operates under.

- **Best Practice knowledge**

  Best practices are shared between the vendor and the client where the client communicates the best ways business processes are executed by mature organizations in their business domain while the vendor shares best ways similar projects are delivered.

- **Business Process knowledge**

  Each organization has a specific way of carrying out its business and this is embedded within its business processes. It is in the interest of the development team to understand the specific nuances of the business and take these into consideration when building a solution. This knowledge, while some of it is documented, mainly resides within the heads of the various domain experts within the organization.

- **System usage knowledge**

  How the business intends using the system and the areas the system will be applied in determines the approach the designers will take in crafting a solution that bests meets the organization's needs. Performance, Security, Availability and Robustness are but some of the decisions that spawn from system usage knowledge.

5.6 Overcoming knowledge sharing challenges

Software development across borders differs from development with co-located teams and as such requires that involved organizations take note of specific challenges that emerge. The signing of an outsource contract does not signify the end of an organizations problems, in fact it

is the start of a journey for the two organizations to understand each other, merge each other's cultures and build bridges across the geographic and language divide.

5.6.1 Diversity of organizations and cultures

Each organization has a personality and the personality is determined by the culture and values of the organization. Organizational culture is what the employees perceive and how this perception creates a pattern of beliefs, values and expectations[104]. Organizational values are the conscious desires that guide the behaviours of its people; generally values are a society's ideas about what is right and wrong. When outsourcing, organizations have to contend with the merging of two distinct organizational cultures, since organizational culture involves shared expectations, values and attitudes, it exerts influence on individual and organizational processes. Approach to quality, work ethic and procedures in dealing with customers are some of the areas where cultural differences between the two organizations become evident. Sharing and organizational culture gives a sense of stability and a single identity between the two organizations, however the tendency to want to export local culture by the client organization to the outsource vendor operation and the tendency by the vendor to want to convert its employees into a client brand usually exposes cultural challenges the two organizations are sometimes ill equipped to deal with. Complicating matters also is the existence of subcultures which exists below the dominant organizational culture, these subcultures form at team, project, division or region level. The subculture can either strengthen the dominant organizational culture or work against it creating conflict and confusion with disastrous effects, especially where management is not aware of its existence. Cultural differences and nuances have a bearing on communication and how knowledge is transferred between the teams, assumptions are made at various levels and things are taken for granted resulting in misunderstandings and potential conflict. The lack of visual contact to understand mood and context sometimes help to compound issues. Lack of knowledge regarding geography specific nuances can lead to requirements not being perceived as anticipated by the client resulting in misdirected effort and rework. The inability of the teams to blend and socialize and therefore share tacit knowledge contributes to knowledge gaps being created. Remuneration and incentive scheme disparities between local and offshore operations in the case of insourcing can contribute towards dissatisfaction and low morale later on in the

[104] Gibson JL et al, 2003

engagement, which can result in staff churn. The discipline of doing things and the difference in the approach to quality by the two organizations can lead to engagement frustrations and inconsistencies in product quality and delivery timelines.

Figure 5.5 - Relationships between the different levels of culture.
(Source: Ribiere V, 2005)

Cultures evolve over a period of time; organizations might therefore not be able to forge a single culture within a short period of time and will need to put an effort in ensuring that the culture of knowledge sharing is nurtured over time. Figure 5.5 illustrates the many cultures that are at play at any given time within an organization. Establishing a formal process of socialization will help the two organizations to expose each other's cultures and facilitate the merger. Socialization is the process by which an organization brings new employees into their culture; it involves the transmission of values, assumptions and attitudes to the new employee[105]. If used correctly,

[105] Gibson JL et al, 2003

socialization can be a good integration strategy for the two teams; however care needs to be taken not to kill the positive elements of diversity where the two teams leverage of each other's strengths.

5.6.2 The nature of software development work

As evidenced in Figure 5.3 above, the nature of software development work being outsourced, unlike in the past is complex and requires that the outsource vendor has a degree of knowledge of the client's business and strategic direction, lack of such domain knowledge results in often extended and costly knowledge transfer sessions with a potential for misunderstandings and later disappointment. When outsourcing, organizations expect to see added value, leverage off economies of scale and have access to strategic business advice[106] and as such the vendor is expected to demonstrate its expertise and bring best practices to the outsource engagement, when this does not happen, the client becomes disillusioned leading to loss of trust and possible termination of the relationship. Organizations are increasingly outsourcing core functions[107] which are the lifeline of their business and as such expect the vendor to provide the necessary security and assurance around such critical services.

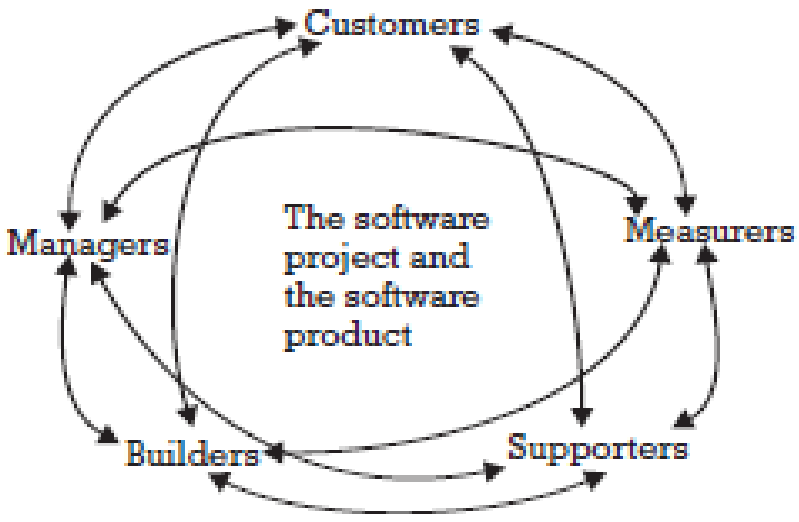


Figure 5.6 - Interaction between groups during software development.

(Source: Evans I, 2004)

[106] Gottschalk P & Solli-Saether H, 2007

[107] Vagadia B, 2007

Figure 5.6 illustrates the interactions that take place between the different stakeholders and the project team with each stakeholder having a different view and different priorities. In this maze of relationships one finds:-

- **The Builders**

  The builders are the technical individuals that make up the solution building team; they are made up of analysts, architects, developers, testers and other subject matter experts. Their priority is to utilize their expertise to build a solution according to agreed requirements and in line with specified budget and timelines. The builders also have to contend with understanding the tools, process and the standards applicable within the project. Builders are on the production line and have to maintain smooth coordination with colleagues both upstream and downstream. While they might be dedicated to the project, in many instances builders also have a responsibility to support production systems in the event of an emergency.

- **The Measurers**

  Measurers have the responsibility of measuring and monitoring the success of the project in line with best practice and agreed metrics. They are the accountants, the quality assurance practitioners and the legal teams who maintain an objective view of project progress; escalating to higher authority when necessary.

- **The Managers**

  Managers are entrusted with ensuring that the promises made regarding the project deliverables are fulfilled and that the project is managed in a transparent and professional manner, they control the planning and management of the delivery process. Managers are made of the project managers, the project sponsor, the business representatives and the steering committee members. Managers are acutely aware of the fact that a big percentage of software development projects do not achieve what they were setup to achieve[108], they therefore have to ensure that their project do not become part of the statistics by keeping a close eye on the actual effort achieved and costs incurred as compared to the budget provided. They have to take decisions regarding the direction of the project and communicate

[108] Standish Group, 2009

status to the stakeholders. Managers bring business knowledge to the project and experience of how things have been in the past, this knowledge they use to guide the project into a path of success.

- **The Customers**

  Customers are the owners of the solution, they are the end users who pay for the solution; they are the ones with a business need the solution being built intends to solve. Customers have the needs of the final customer to meet and as such any delay of the project has a direct impact on their plans and strategies. The customers specify the requirements of the system and test to provide final confirmation that the system is in line with their requirements. Customers bring business context to the project; the system is intended to automate their business processes, therefore they will evaluate the system on how efficient it allows them to execute their tasks.

- **The Software Project and the Software Product**

  The software project is a vehicle for delivering the software product. Knowledge from various experts involved and the artefacts used in the project ensure that the software product is delivered according to the customer's expectation. The software product is revised and modified during the life of the projects to meet changing business needs. Once completed, the software product becomes part of the organization's knowledge artefacts.

While the work of the different players is collaborative in nature, the pressures, competing priorities and disagreements on approaches create a force that make it difficult to prioritize knowledge management activities and ensure that knowledge created from the project is harnessed and managed in a manner that makes it utilizable to the wider organization.

5.6.3 Investing in technology

Outsource deals are struck to last over a number of years and therefore there is a need for the parties to invest in processes and technologies and other related infrastructure to enable the teams to transact efficiently across the geographic divide. The investment can take the form of digital communication lines, video conferencing equipment, language laboratories, secure development areas and branding paraphernalia to encourage a cross cultural fit. Investment requires long term commitment as such both parties might be reluctant to commit until the relationship has proven

itself, such delays might have an impact on how the two teams relate to each other and therefore share knowledge across the divide. Once an investment is made and the two centres are connected, the possibilities become endless and the two teams can collaborate seamlessly across the divide with little effort. Establishing access to the organization's intranet site in order to allow the offshore team to keep track of activities within the organization will be one way of fostering a sense of belonging for the distant team, on the other hand such facilities might be perceived to undermine the security of the internal network and may therefore come under constant review.

5.6.4 Creating incentives to share knowledge

Employees will not necessarily share knowledge out of the goodness of their hearts; relying on people acting altruistically does not guarantee that knowledge will be shared. There are several reasons why employees will not participate freely in knowledge sharing activities, these include:-

- **Volume of work**

  As indicated in Fig. 5.5, software development project is a stressful, high pressure environment, the sheer volume of work allocated to each resource often make it impossible for employees to have time for sharing knowledge. The activity of documenting one's experiences may not be the most exciting thing to do and as such can end up being relegated to a lower priority. Pressure to come under budget implies that activities outside of the project scope cannot be funded and such knowledge management activities often become out of scope items as a result rather than spend time on these tasks, employees choose to focus on core project activities. The idea that knowledge sharing activities constitute free time or are regarded as out of scope items diminish its value and create a perception of less importance amongst project participants and employees in general.

- **Loss of status**

  Knowledge is power and so is experience; those who possess it wield power over those who don't[109]. When an expert "gives away" his knowledge through sharing, he can perceive this to be lowering his value and negatively influencing his status within the organization; as a result he would be inclined to hold back on sharing knowledge, leveraging it instead for

[109] Pettigrew A, 1972

financial gain. An expert might hold back in sharing knowledge as a bargaining tool against retrenchments in times of restructuring and cost cutting. While this might only be a perception, it is one that needs to be countered as it undermines the efforts of sharing knowledge within the project and the organization as a whole. To counter this barrier, the workload of experts need to be reviewed and they need to be given time to attend to knowledge sharing activities, assurance also need to be given regarding the importance of knowledge sharing within the organization and the esteem in which people who share knowledge are held. Management commitment to knowledge sharing needs to be visible through support and actions. While volunteering is a good gesture, to succeed; knowledge management cannot be based purely on volunteering; it has to be part of an organization's operational requirements.

- **Unhealthy competition**

  In an organization where the structure is designed to encourage competition amongst its employees or divisions, employees will tend to hold back on their knowledge and not share with those they regard to be on the opposing side. Such structures set employees against each other creating an atmosphere of unhealthy competition. Employees end up being in a tournament where the last man standing gets the trophy, this results in employees holding back on sharing and leveraging their knowledge to position themselves at the top of the pile. As indicated above, knowledge is power and therefore sharing it might be perceived to empower those who could end up using the knowledge against the sharer. Organizations could remedy this by ensuring that unhealthy competition is discouraged and by affording status to those who are seen to be helping others through sharing their knowledge.

- **Lack of confidence**

  An organization might not recognize an employee with expertise at a particular domain and as such, such employee may innocently hold back on knowledge that could help the team. It could also be that he lacks the confidence to display his knowledge in public. Management need to be aware of such cases and create an environment where all employees would be keen to share their knowledge.

- **Lack of trust**

  Knowledge reuse is meant to reduce the effort of performing a task and to improve quality as less steps are performed, however if the repository on which is based is not trusted by potential users, it will not be utilized. Lack of trust may come as a result of employees questioning either the content or the quality of the knowledge held in the repository; the way the knowledge is processed and stored may also make it difficult for others to access and use. Organizations need to ensure that there are quality assurance and control processes in place to guarantee the validity and ease of use of knowledge held in their repositories. Part of the quality assurance process is to validate the knowledge and ensure suitability before capturing it into the organizational repository.

- **Difficulty in articulating tacit knowledge**

  Tacit knowledge is intangible knowledge accumulated unconsciously by an expert, it is not documented in any form of artefact and as a result of that requires a skill to share. Sharing such knowledge might even be more difficult with novices who lack the context of where the knowledge originates from. Knowledge flows better under ideal conditions where relationships and trust are encouraged and context is communicated, as a result knowledge sharing might not occur effectively where sharing is enforced and governed through rigid rules. Organizations need to invest in training interventions to educate their experts on ways of articulating tacit knowledge; these could be presentation skills and public speaking.

5.7 Reducing dependency through knowledge retention

With a key part of the organization's activities being performed outside of its control, an organization needs to develop strategies to retain core knowledge to maintain competency and to reduce risk in the event of the relationship going sour. Big organizations like IBM, Microsoft and Google have " invaded" traditional outsource locations such as India establishing big development centres that dwarf local players, this has fuelled competition for talent leaving smaller organizations at a risk of losing to the big conglomerates who have a pulling power because of their deep pockets and established brands.

Intellectual property theft poses a big danger to an organization's survival, if not properly managed the organization's competitive advantage can fall into the hands of the competition or

can be compromised by the departure of a vendor's key resources. The challenge of retaining knowledge in the outsource relation is that knowledge location is in a constant flux oscillating between the vendor and the client with each constantly acting as a provider and a consumer at the same time. The distributed nature of outsourced software development makes it difficult to physically locate the latest versions of knowledge artefacts such as source code and task descriptions resulting in inconsistent development processes and missed reuse opportunities[110].

Incentive	Eligibility	Impact
Equity and equity linked options	Middle and senior management.	Very high – given pre-IPO (initial public offering) nature of most KPOs.
Technical focus (leads to staff acquiring advanced globally accredited certifications)	2 to 5 years experience and relatively junior analyst staff.	High – as the company is seen to be committing to developing its staff into globally recognized professionals.
Promotion into relationship management roles	3 to 7 years experience.	High – as the company is seen to be committed to all-round professional growth.
Annual bonuses (20% to 25% of base salary)	All staff.	Low – has become industry standard.
Sign-on bonuses (10% to 20% of base salary)	Staff from tier 1 graduate, post-graduate and professional programs.	Low – has become industry standard.
Incentives such as overseas trips and on-shore client work	About 25% of staff in captives and 5% in third-parties.	High – as staff normally want to move from back office roles to front-end client facing roles.

Table 5.2 - Skills retention strategies for Knowledge Processing Organizations.
(Source: KPMG, Swamy & Associates, 2008)

Table 5.2 shows the different incentive strategies that Knowledge Processing Organizations use to retain critical skills. The strategies offer varying incentive forms calculated to appeal to the different levels of employees in an effort to ensure retention of knowledge. The strategies need to be reviewed and revised regularly as the retention effect diminishes as competition reciprocates by providing similar or better choices. As can be seen from the above analysis, developing software across geographically dispersed locations is a tug of war between the pushing forces of distance and the pulling and counter-balancing forces of standards, compliance and consistency, these pushing forces manifest themselves in the form of coordination breakdown, loss of communication richness, cultural differences and lack of a cooperative spirit while the counter-

[110] Meyer B et al, 2007

balancing forces include product architecture, ICT infrastructure, collaboration technologies such as email and video conferencing, Management techniques and Delivery methodologies and standards[111]. The bottom line is that knowledge sharing does not occur spontaneously, to succeed it requires active intervention and incentivization on the part of the organization.

5.7.1 Offshore insourcing as a knowledge retention strategy

While in the beginning detractors of outsourcing were concerned with potential job losses emanating from massive outsourcing, studies have however shown this not to be the case. The real concern that has begun to surface from outsourcing is the loss of critical organizational competency in knowledge centric roles that are key to driving corporate strategy[112]. As described before, insourcing involves the use of the organization's IT department or its subsidiary as a provider of IT services, in the case of offshore insourcing, the service provider is located outside the organization's geographic borders, most commonly overseas. Organizations apply different decision models to help them determine the best mix of services to be insourced and where to locate these insourced services, see Figure 5.7 below. Economics and Risk are the key drivers of such decisions and the decision model in Figure 5.7 helps to surface all the issues and assists in making the right decisions.

[111] Carmel E, 1999

[112] Karamouziz F & Young A, 2004

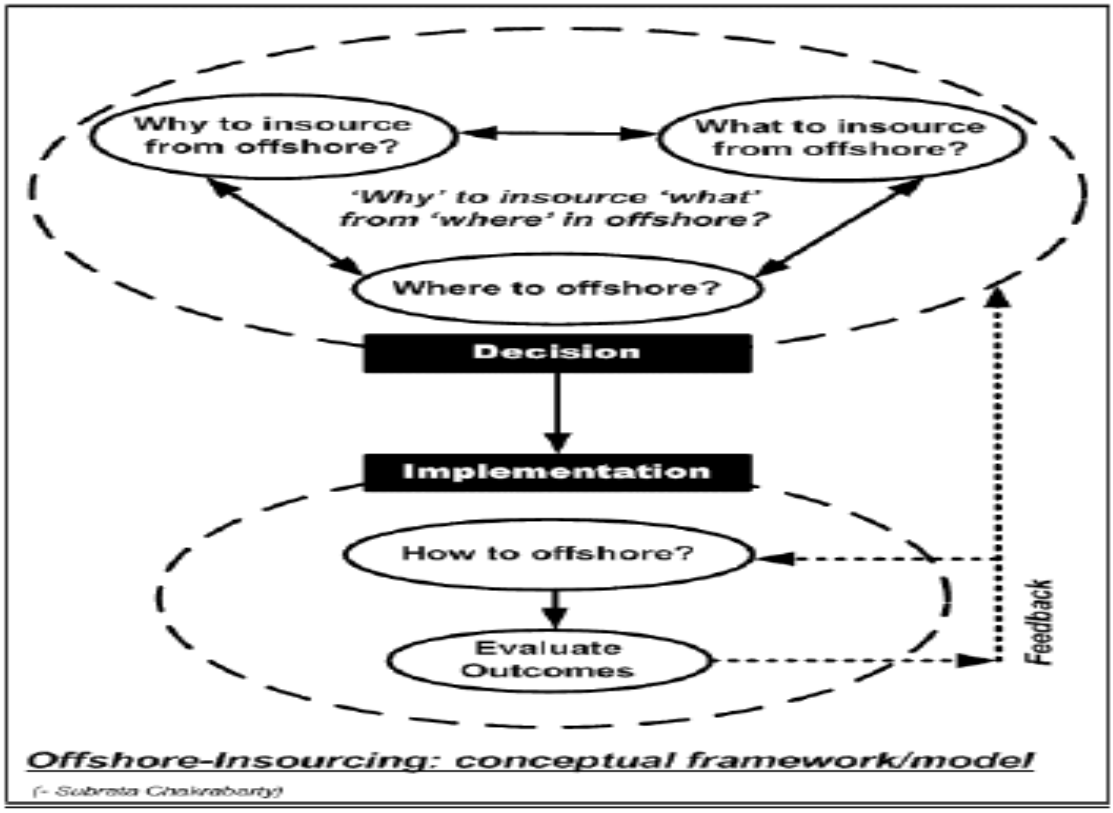


Figure 5.7 - Conceptual framework for Offshore insourcing.
(Source: Chakrabarty S, 2006)

Outsourced software development projects have a finite engagement timeline as a result the engagement soon terminates and the teams move on to other projects. Knowledge accumulated risks being lost or landing in wrong hands as a result of staff migration. Realizing the potential loss of knowledge to competition as a result of outsourcing and the effect this could have on the organization's commercial viability, organizations are using a variation of outsourcing called offshore insourcing as an option to gain access to low cost high quality skills of offshore IT professionals while retaining critical competencies and knowledge centric roles within the organization, but not necessarily within the nation[113]. Offshore insourcing therefore attempts to capitalize on the benefits of outsourcing while mitigating the risks of offshoring. Through offshore insourcing an organization would acquire or establish a software development entity through which it will offshore software development activities, the advantage of this strategy is

[113] Chakrabarty S, 2006

that offshore resources are "owned" by the company and therefore subscribe to the culture, policies and values of the organization involved. The interaction between the various elements of the organization is not limited only to the IT department, making knowledge sharing efforts more widespread and integrated. The power of Information and Communication Technology (ICT) to transcend the limitations of time and space[114] has made the offshore insourcing model to be both viable and practical for organizations wanting to establish an insourced operation.

Overall insourcing gives an outsourcing organization the ability to have greater control over offshore resources, better control of its intellectual property and less resistance from staff as they view the offshore operation as part of the organization's expansion strategy and therefore part of their own. Offshore insourced operations face the same challenges as normal outsourcing, the difference is that the client organization owns the processes and structures that deal with the resolution of these issues and is therefore in a better position to influence and align them. To test the offshore insourcing waters and ensure a smooth transition, organizations would normally employ a Build Operate Transfer (BOT) strategy to establish the offshore insource centre where a credible vendor would be assigned the task of establishing, operating and maturing the centre for a while before transferring control back to the insourcing client. The advantage of this model is that it gives the client organization the ability to "test drive" the model under the guidance of an experienced vendor and to take time familiarizing themselves with the challenges and nuances of outsourcing. Gartner uses the table below to provide a view of the differences between offshore outsourcing and offshore insourcing[115].

[114] Carmel E, 1999

[115] Scholl R & Chohan S, 2003

Two Approaches to Outsourcing

OffShore Insourcing	Offshore Outsourcing
Advantages	**Advantages**
Global enterprises can get significant cost benefits from centralizing processes in a geographic location	Better economies of scale and therefore, lower costs
Process control remains in-house	Access to better and continually enhanced technology
Process knowledge remains with the organization	Inherent incentives to improve the process
Better security	Ability to benchmark to industry and global standards
Faster turnaround of processes	
Limitations	**Limitations**
Higher costs compared to outsourcing	Less control over process delivery
Slower response to change	Presently limited to specific functions within a business process
Lack of motivation to improve processes	Still nascent and unproven service offerings
Initial overtime cost savings become the new "legacy," and the next level of cost savings can only be achieved through the introduction of improved technology/processes or the transition to a multiclient shared service center	Limited industry and process knowledge (although it is growing)
Risks of transferring inherent inefficiencies to the insourced center	Relationship management over the long term is crucial to successful outsourcing
Commitment to manage remotely located delivery center is intense	

Table 5.3 - Approaches to outsourcing.

(Source: Gartner Research, 2003)

As can been observed from Gartner's Table 5.3 above, both Outsourcing and Insourcing have their advantages and limitations. Each option is better suited to certain strategic and marketing conditions and organizations need to evaluate their needs in order to decide which would work better for them. While insourcing will mitigate the risk of knowledge pilfering to a certain extent and provide better security for the organization's intellectual assets, it tends to be more costly than outsourcing and is sometime bogged down with the organization's internal politics. On the other hand outsourcing brings agility, costs advantages and access to enhanced technologies without the high price of investment; these advantages are however countered by lack of control, limited industry knowledge and a need to closely manage the relationship. Depending on the

needs of an organization, some organizations will adopt a best of breed approach where a range of niche outsourcing deals are created with different specialist service providers in order to address specific needs that might not be addressed by a generic vendor and to spread the risk of failure.

5.8 Managing shared knowledge

During project delivery, expertise from the vendor and client organizations combine to yield new knowledge, this project memory created over the duration of the engagement is an asset the client organization needs to ensure that it is retained. Without structures to identify, capture and retain project memory it would simply be lost into the minds of those involved resulting in dependencies and exposure due to single points of failure. High dependency on a particular vendor may lead to prohibitive costs should the client organization want to switch to a different, more desirable vendor[116]. The loss also can result in the organization not being able to reuse generated knowledge in other scenarios that may occur in the future.

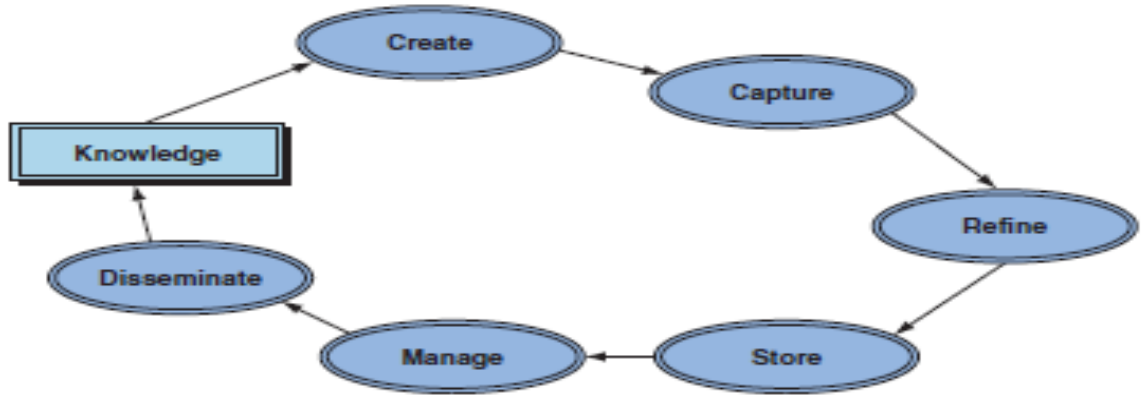


Figure 5.8 - The knowledge management lifecycle

(Source: Huysman M & De Wit D, 2003)

A structured knowledge management strategy in line with the activities highlighted in Fig5.8 needs to be put in place to ensure that all created knowledge artefacts are captured, refined and stored for reuse. The captured knowledge needs to be disseminated to the relevant project teams

[116] Davis-Blake A & Hui P, 2003

and a management process needs to be put in place to ensure that knowledge is reviewed for currency and relevance.

5.8.1 Organizing for effective knowledge flows

The way the delivery team is structured influences the way it communicates within itself and the effectiveness of its governance processes. Even with the most elaborate knowledge management strategy in place, if the team is not structured for efficient knowledge flows, knowledge will continue to leak out and the leverage and reuse opportunities will be missed.

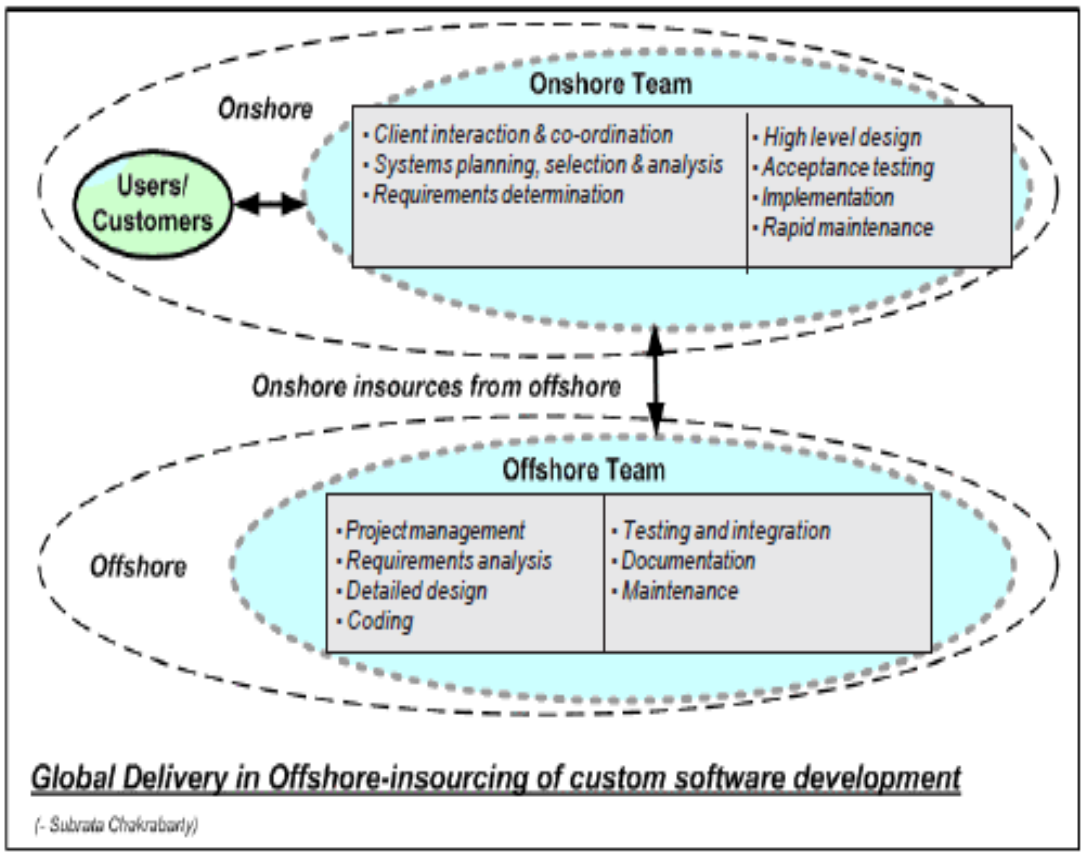


Figure 5.9 - Offshore/Onshore delivery team structure.

(Source: Chakrabarty S, 2006)

As illustrated in Figure5.9, the delivery team needs to have management structures on both sides of the shore to ensure that agreed processes and standards are observed.

Key to this global software delivery structure would be the following principles:-

- Knowledge intensive activities are performed onshore closer to the business while factory type activities are moved offshore and executed by low cost resources.
- There are established communication links between the two teams to encourage sharing of information and lessons learnt.
- A knowledge repository accessible and updatable by both sides is established.
- There is constant movement and rotation of resources between the teams to exchange cultures, increase collaboration and to minimize misunderstanding.

All the above features facilitate the smooth flow of knowledge ensuring the effectiveness of the organization's knowledge management strategy. ICT collaborative tools like email, video conferencing and instant messaging and knowledge portals are used to bridge the distance and ensure an enhanced communication experience[117]. The model allows for an organization to have several global delivery centres based on a similar structure, coordinating according to the same principles and greatly increasing delivery capability across time zones while lowering costs.

5.8.2 Managing knowledge as part of a corporate strategy

So far we have highlighted the importance of knowledge management to an organization, the processes that drive knowledge management and barriers to the implementation of an effective knowledge sharing culture. There is however a more critical role of knowledge management and that is the criticality of knowledge management to the attainment of an organization's strategic goals, this implies that all the knowledge management activities mentioned above need to be underpinned by a knowledge management strategy which is aligned to the organization's competitiveness objectives. Having an appropriate knowledge strategy in place is essential for ensuring that knowledge management efforts are being driven and supported by and are supporting the organization's competitive strategy[118]. The presence of a knowledge management strategy indicates its belief that knowledge is a strategic resource that is essential for its long term sustainability. In an outsource scenario, the presence of a knowledge management strategy means that both organizations are able to refer to the strategy for alignment, they can ensure that

[117] Carmel E & Agarwal R, 2002
[118] Zack M, 1999

their knowledge sharing activities support the knowledge management strategy and that the behaviour of all team members involved in the project will be checked against the prescriptions of the strategy.

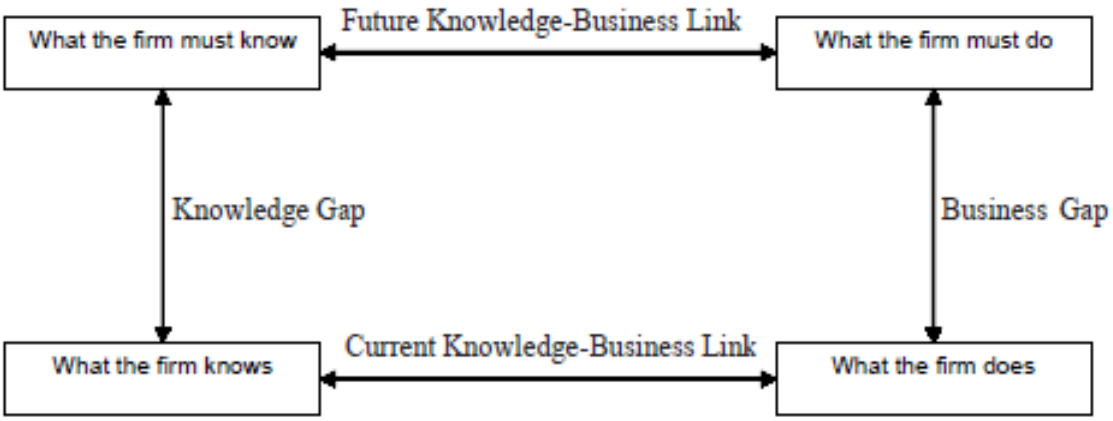


Figure 5.10 - Knowledge gap as derived from the Business gap.
(Source: Gottschalk P, 2005)

As highlighted in Figure 5.10, it is important for organizations to ensure that their knowledge strategies align with their business strategies and that the business strategy compliments the knowledge strategy. When an organization sets its vision for the future, it does an assessment of its current capabilities against the competencies required to deliver its target state, it therefore implies that a gap analysis needs to be done on its current knowledge to determine what the organization must know to deliver on its strategic goals. Executive support for the knowledge management strategy implies that it will be easily enforceable at all levels and that it will get the necessary support and resources to enable it to prosper. When the organization's knowledge management capabilities have matured, the executive sponsorship role for knowledge management would be assumed by the Chief Knowledge Officer; a senior executive reporting directly to the Chief Executive Officer.

The Chief Knowledge Officer would amongst other things be responsible for[119] :-

- Being the champion for knowledge management within the organization.
- Manage the organization's knowledge management competency including its staff and plans.
- Manage relationships with external service providers.
- Provide strategic input to the executive regarding the direction of knowledge management within the organization.
- Measure, manage and promote the value of knowledge management contribution to the organization's strategic direction.
- Architect the organization's knowledge management strategy, processes and tools.

Figure 5.11 - A typical knowledge management strategy framework.
(Source: Gottschalk P, 2005)

A knowledge management strategy framework (see Figure 5.11) can be used to position knowledge management within an organization and align it to the organization's strategic intent.

A typical knowledge strategy framework would consist of the following key components:-

[119] Davenport TH & Prusak L, 1998

- **The Strategy**

  This constitutes the alignment area where the knowledge management strategic objectives and value proposition are articulated.

- **Culture and Behaviour**

  This is where the expected organizational culture and behaviour is articulated in order to foster an environment of free flowing knowledge. As indicated before; culture influences behaviour and knowledge sharing can be facilitated or hindered depending on the culture that is prevalent within the organization.

- **Structure and Processes**

  This constitutes the knowledge management operating model and governance, all processes regarding knowledge Discovery, Capturing, Sharing and Utilization would be articulated here.

- **Metrics, Results and Rewards**

  This constitutes the measurements on the adoption and maturing of knowledge management within the organization, achievements of set goals and rewards and incentives to encourage buy-in.

5.9 Security considerations in offshore knowledge sharing

While the benefits of offshore outsourcing might be attractive, organizations need to ensure that their knowledge management strategies include the securing of knowledge assets. Regardless of the number of contracts signed with service providers and the reliability of the service providers chosen, the responsibility of guaranteeing an organization's knowledge assets rests with the organization itself. Below are some of the security challenges, organizations face when implementing offshore software development outsourcing.

5.9.1 Securing the extended enterprise

Computer security has become a major concern for all businesses over the last few years; this is as a result of an exponential increase in computer related crime ranging from hacking, defacing of web sites, denial of service attacks and credit card related fraud. Criminal syndicates with a global footprint sometimes working in collaboration with internal employees are defrauding

companies and unsuspecting members of the public of millions of dollars. The internet has become a risky place to do business in with computer viruses and worms becoming more destructive and sophisticated hacking tools freely availability for everyone to download. The threat of being compromised by these security breaches has forced companies to spend millions of dollars protecting their networks and systems. Hacking is constantly evolving making the fight to stop it an on-going challenge. The interconnection of the global village means that an organization cannot be immuned from the security challenges experienced by the rest of the world. Security is not a matter for the IT department alone but a responsibility of the whole organization.

The collaboration between an outsourcing organization, the service provider and vendors through which a service provider might subcontract some of the work create an extended enterprise with added vulnerability points. As the enterprise extends even further, these vulnerabilities become difficult to monitor and manage as they are outside the organizations traditional scope of control and influence. The degrees of process and governance maturity may differ wildly between the components that make up the extended enterprise making it difficult for an organization to assess with certainty the level of compliance across the value chain. While an organization might be governed by strict compliance laws such as King III, Sarbanes Oxley and other consumer protection legislation, the service provider and other participants in the value chain might not necessarily come under such laws and as such may not meet the prescribed compliance requirements.

Once an organization's intellectual property leaves its domain of control, it becomes the responsibility of the service provider to ensure its continued security and as the organization begins to rely on its partners for the security of its knowledge assets, it has to assure itself on whether the service provider has the necessary credential to be trusted with such a responsibility. There are several areas where an organization's information and knowledge assets can be compromised when operating within an extended enterprise, these include:-

- **Theft**

  Theft can be of the identities of internal resources where such information would be used by outsiders to hack into an organization's systems with the intention of causing harm.

Theft can also be intellectual property relating to business processes, new products going into the market or the security architecture of an organization's systems. This information can be sold to the competition undermining an organization's competitive urge or it could land up in the hands of hackers thus compromising the organization's security measures and damaging its image. Theft can also be of client information which can be used for such crimes as forgery, black mail and extortion. This information can also be used to hack into client accounts resulting in law suits and reputational damage.

- **Misinterpreted requirements**

  With so many links in the value chain, it is possible that as requirements filter down from service providers to sub-contractors, requirements can end up being misinterpreted causing a system to execute outside of the intended scope. While these will normally be caught in the process of testing the solution, there is likelihood that some of the bugs may find their way into an organization's production systems.

- **Cultural differences**

  Within a global economy, it is possible that an organization can outsource its software development to India and the Indians in turn can subcontract part of the work to China and other parts to Venezuela. Communication breakdown as a result of unchecked cultural differences between the teams involved could result in missed deadlines or a possibility of rework and escalating delivery costs.

5.9.2 Using the International Information Systems Security Certification Consortium (ISC) ² guidelines to define IT security

The International Information Systems Security Certification Consortium (ISC) ² has defined an information systems security body of knowledge which consists of ten domains from which security professionals can be accredited with a Certified Information Systems Security Professional (CISSP) certification. The certification gives the organization an assurance that its IT security personnel are well versed with matters of protecting the organization's information assets.

When outsourcing to a vendor, organizations are entrusting their most valuable asset, their knowledge to a third party, it is important therefore to ensure that the vendor subscribe to an accredited framework with regards to protecting the IT Systems and knowledge they are entrusted with, to do this; an organization would need to test the vendor on their application of the CISSP principles. Each of the (ISC) ² domains (refer to table 5.5) deals with a specific area of systems security and together the domains complement each other ensuring that the security processes put in place are such that the security of an organization's IT systems is guaranteed. The IT security personnel who are CISSP certified maintain an ethical standard that is in line with the responsibility of their roles.

Implementing security controls is part of an organization's strategy for managing risk and ensuring availability of service to its clients, failure to do this can permanently harm the business and its brand rendering it incapable of recovering. Organizations have to classify the threats that they face and formulate strategies that deal effectively with each of those threats, risks identified have to be owned and the owners of the risks have to attest that they understand the status of their risks at all times and are comfortable with such. The cost of security failure is far greater than the protection required to prevent it from happening; therefore organizations must ensure that they are not penny wise and pound foolish.

Domain	Description
Access Control	This domain examines mechanisms and methods used to enable administrators and managers to control what subjects can access, the extent of their capabilities after authorization and authentication, and the auditing and monitoring of these activities. Some of the topics covered include: • Access control security models • Identification and authentication technologies and techniques • Access control administration • Single sign-on technologies • Attack methods
Telecommunications and Network Security	This domain examines internal, external, public, and private communication systems; networking structures; devices; protocols; and remote access and administration. Some of the topics covered include: • OSI model and layers • Local area network (LAN), metropolitan area network (MAN), and wide area network (WAN) technologies • Internet, intranet, and extranet issues • Virtual private networks (VPNs), firewalls, routers, bridges, and repeaters • Network topologies and cabling • Attack methods
Information Security and Risk Management	This domain examines the identification of company assets, the proper way to determine the necessary level of protection required, and what type of budget to develop for security implementations, with the goal of reducing threats and monetary loss. Some of the topics covered include: • Data classification • Policies, procedures, standards, and guidelines • Risk assessment and management • Personnel security, training, and awareness
Application Security	This domain examines the security components within operating systems and applications and how to best develop and measure their effectiveness. It looks at software life cycles, change control, and application security. Some of the topics covered include: • Data warehousing and data mining • Various development practices and their risks • Software components and vulnerabilities • Malicious code
Cryptography	This domain examines methods and techniques for disguising data for protection purposes. This involves cryptography techniques, approaches, and technologies. Some of the topics covered include: • Symmetric versus asymmetric algorithms and uses • Public key infrastructure (PKI) and hashing functions • Encryption protocols and implementation • Attack methods

Domain	Description
Security Architecture and Design	This domain examines concepts, principles, and standards for designing and implementing secure applications, operating systems, and systems. This covers international security measurement standards and their meaning for different types of platforms. Some of the topics covered include: • Operating states, kernel functions, and memory mapping • Enterprise architecture • Security models, architectures, and evaluations • Evaluation criteria: Trusted Computer Security Evaluation Criteria (TCSEC), Information Technology Security Evaluation Criteria (ITSEC), and Common Criteria • Common flaws in applications and systems • Certification and accreditation
Operations Security	This domain examines controls over personnel, hardware, systems, and auditing and monitoring techniques. It also covers possible abuse channels and how to recognize and address them. Some of the topics covered include: • Administrative responsibilities pertaining to personnel and job functions • Maintenance concepts of antivirus, training, auditing, and resource protection activities • Preventive, detective, corrective, and recovery controls • Standards, compliance, and due care concepts • Security and fault tolerance technologies
Business Continuity Planning (BCP) and Disaster Recovery Planning (DRP)	This domain examines the preservation of business activities when faced with disruptions or disasters. It involves the identification of real risks, proper risk assessment, and countermeasure implementation. Some of the topics covered include: • Business resource identification and value assignment • Business impact analysis and prediction of possible losses • Unit priorities and crisis management • Plan development, implementation, and maintenance
Legal Regulations, Compliance, and Investigation	This domain examines computer crimes, laws, and regulations. It includes techniques for investigating a crime, gathering evidence, and handling procedures. It also covers how to develop and implement an incident-handling program. Some of the topics covered include: • Types of laws, regulations, and crimes • Licensing and software piracy • Export and import laws and issues • Evidence types and admissibility into court • Incident handling
Physical (Environmental) Security	This domain examines threats, risks, and countermeasures to protect facilities, hardware, data, media, and personnel. This involves facility selection, authorized entry methods, and environmental and safety procedures. Some of the topics covered include: • Restricted areas, authorization methods, and controls • Motion detectors, sensors, and alarms • Intrusion detection • Fire detection, prevention, and suppression • Fencing, security guards, and security badge types

Table 5.4 Security Domains that make up the CISSP body of knowledge.

(Source: Harris S, 2008)

Policies are an integral part of managing risks and ensuring a secure risk free computing environment. A security policy is an overall general statement produced by senior management or a selected policy committee that dictates what role security plays within the organization, it sets out goals, assigns responsibilities, dictates enforcement and guides the sanctioning of transgressors[120]. Key security considerations to be observed within an outsource relationship include:-

- Trustworthiness of vendor staff.
- Regulatory compliance.
- Confidentiality of records and management of output material.
- Security of application designs.
- Enforcement of standards and management of controls.
- Security of the communications infrastructure between the locations.
- Management of access controls and surveillance at outsource centres.
- Completeness of disaster recovery and business continuity plans.

Depending on the threats assessed and appetite for risk, organizations will implement different levels of security in their internet architectures to protect the enterprise against intrusion.

5.10 Ensuring effectiveness of controls

Control measures as articulated above are meant to surface and mitigate risks that have been identified, they are meant to provide management and stakeholders with comfort that the outsourcing operation does not expose the organization and its knowledge assets to undue risks and that whatever risks that can be identified are successfully mitigated. There is however a danger that with all the controls that have been put in place, disaster can still strike and organizations can suffer irreparable damage, it is for this reason that monitoring, assessment and regular testing of controls is required to ensure their effectiveness and relevance under changing business conditions.

[120] Harris S, 2008

5.10.1 Control Self Assessments (CSA)

Control self-assessment is a methodology used to review key business objectives, risks related to achieving these objectives and the controls designed to manage those risks[121]. The key aim of CSA is for an organization to take ownership of risk controls, ensuring that employees and strategic partners such as outsource service providers are aware of them and that they comply as required. Observations made in the assessment exercise refer to areas where it is found that certain controls are either ineffective or non-existent; in which case the control owner needs to put in place measures to address such findings. The advantage of implementing Control Self Assessments is that they are internally driven by management and are performed with the pressures of auditors wherein if a finding is made it will be escalated to the executive committee. Control Self Assessments give management the opportunity to resolve control issues before they become an audit issue which could have a negative effect to the organization's reputation. An organization can also request the CSA of its service provider to satisfy itself of management commitment as well as the health of the controls.

5.10.2 Disaster Recovery and Business Continuity plans

While software development might not affect the business's day to day operations directly, disaster in the software development environment can set back an organization's plans to deliver a strategic product compromising its market dominance. The development environment is also used to resolve issues that emanate from production systems, a non-functional development environment will result in a production issue not being resolved on time leading to irate customers and lost business opportunity. In certain instances, the software development environment functions as a backup to the production environment in the event of a disaster, its availability to perform as such needs to be tested periodically and confirmed to be ready. The service provider needs to assure management about its disaster recovery and business continuity plans. These need to be tested regularly and results of the tests kept as proof of its functioning.

[121] Information Systems Audit and Control Association (ISACA), 2010

5.10.3 The role of IT Governance

Governance is a process whereby senior management exerts control over business functions through policies, objectives, delegation of authority, and measurement[122]. Governance requires that an organization has a functioning structure through which tasks are executed and decisions taken. The service provider needs to be assessed in line with its governance structure and the quality of the people that serve in the decision making process. COBIT is a best practice IT governance framework upon which many progressive organizations base their governance structure. To demonstrate efficiency of its governance, organizations needs to maintain a balance score card which highlights all the key objectives being tracked, the key measures used to indicate success and the performance of the organization against such targets.

5.11 Conclusion

Globalization has caused companies to revise their business models, the proliferation of IT infrastructure, the availability of affordable IT skills in locations such as India and China and the abundance of broadband connecting the different locations of the world has resulted in the viability of the outsource option providing cost benefits of significant magnitude. Offshore outsourcing by its nature results in organizations having to share knowledge across geographically dispersed locations, they can choose to either keep IT service delivery internally (insourced) or they can outsource it to any of the third party vendors that specialize in the provision of IT services, in both options the service can either be provided onshore or it can be offshore. Outsourcing can be further broken down into more permutations where the partnership can range from building capability with the intention of insourcing later or a long term strategic partnership where the services will be expanded as the relationship matures. Regardless of the option chosen, the key principle of outsourcing is to ensure that the partnership is a win-win arrangement with both parties deriving equitable benefit from the relationship.

While in the beginning outsourcing was confined to low level jobs, the trend has now shifted towards more high value jobs being moved offshore with organizations taking advantage of the abundance of affordable, highly skilled resources.

[122] Gregory P, 2010

When fully operational the outsource relationship results in a maze of knowledge flows between the two organizations, these knowledge flows emanate from the new knowledge being created as part of delivering the project as well as the constant sharing of knowledge that occurs between the various team members and other stakeholders both locally and offshore. As illustrated before, all stages of the Software Development Lifecycle contribute towards knowledge creation; a knowledge sharing strategy formalizes the handling of knowledge shared in each of the SDLC stages and maximizes the benefit of outsourcing to an expert organization. Knowledge generated during project delivery differs in context and application, it is therefore important that it be categorized in order to manage it properly and to ensure that it is provided the necessary classification and security where necessary. With a functional knowledge sharing strategy, the vendor team is quickly on-boarded and they begin to deliver value immediately fast tracking the project and saving the organization valuable time and money.

While sharing knowledge is a strategic imperative to the success of a project, there are however barriers to knowledge that partnering organizations need to observe and overcome lest the project be jeopardised. Barriers such as the organizational culture, work ethic, the work stream mentality can create false associations and rally employees along a distractive cause creating divisions and undermining the culture of collaboration, management needs to ensure that this is identified and addressed early. Sharing knowledge doesn't necessarily occur naturally among team members, employees may view their knowledge as a lever to get recognition and reward, to others knowledge might be a source of status, therefore sharing their knowledge might make them feel disempowered. Management need to provide targeted incentives that will motivate team members to share their knowledge without feeling pushed. Once shared knowledge needs to be retained to ensure that it is accessible to others and that dependency on those who own the knowledge is mitigated. Where knowledge security is a major concern, in-sourcing could be utilized ensuring that knowledge will be handled by internal resources regardless of where they might be located.

The way teams are structured influence how knowledge will flow within the project, the structure should encourage contact and collaboration and tools need to be in place to bridge the geographic divide and foster collaboration. To ensure that knowledge management is sustainable both within the project and in the organization as a whole, it needs to be underpinned by a knowledge management strategy that is aligned to and is a subset of the wider enterprise

strategy. Knowledge management needs to be driven from the top; preferably by an executive such as the Chief Knowledge Officer. The knowledge management strategy sets the rules regarding the vision and structure of knowledge management within an organization, the supported behaviour and the measurers put in place to track success and potential benefits to both the organization and the individuals. Knowledge management strategy also provides direction and general guidelines on the management and maturing of knowledge assets. Without a proper security strategy, all the knowledge management ideals will not materialize, security plays an even bigger role when outsourcing to offshore third parties. The International Information Systems Security Certification Consortium framework can be used to ensure that security standards are met and that those tasked with managing the IT security of the enterprise are capable and operating in line with best practice. The domains of the (ISC)² cover all the areas critical to ensuring a secure outsource environment.

This chapter showed how pertinent sharing of knowledge is to the success of the outsource partnership, it also addressed the barriers that inhibit sharing, where these emanate from and how they can be addressed in order to open knowledge flows, there was also highlight of the risks involved in sharing knowledge and how these could be successfully mitigated with well-designed interventions. Perhaps the key point to be made is the strategic role that knowledge plays in enabling the organizational strategy making a case for knowledge management to be located at senior management with an executive sponsorship.

The next chapter concludes by reflecting on the picture that has been painted highlighting the importance and benefits of knowledge in enabling successful delivery of outsourced software development projects, acknowledging limitations that are inherent in research of this nature and examines future areas where this research can be expanded to further enhance available knowledge on the subject.

Chapter 6
Concluding findings and Recommendations

6.1 Research objectives achieved

This research study set out to establish the significance of the role played by knowledge management in the success of offshore outsourced software development projects. Linked to this objective was a need to highlight key knowledge management strategies that organizations can utilize to safeguard their knowledge assets when outsourcing software development. The hypothesis advanced at the beginning of this study was that; if software development is outsourced offshore then knowledge management would be a critical factor in the successful delivery of such software projects. Disproving this hypothesis would imply that organizations could guarantee the success of their offshore software projects and secure their knowledge assets without paying attention to knowledge management. Proving the hypothesis would quantify the significance of knowledge management strategies in offshore software projects thereby answering the questions posed by the research problem.

To validate the hypothesis the study highlighted the following areas where knowledge management was decisively influencing the success of outsourced software development projects:-

- Project success through the management of SDLC knowledge.
- Utilizing Knowledge Management Systems to minimize project risks.
- Using knowledge to maintain project control across borders.
- Securing organizational intellectual property.
- Making outsourcing succeed as a business strategy.

.

6.1.1 Project success through the management of SDLC knowledge

By analyzing the knowledge generation capabilities of each of the SDLC stages, the research has shown that software development lifecycle is a source of organizational knowledge. The requirements and system documentation of the SDLC provides knowledge externalization which would not normally be formalized, the criticality of this process is in capturing tacit knowledge into forms that can be easily shared. Reference to the Standish Group report highlighted the fragile nature of software development projects and the fact that a large percentage fail to deliver. The research contends that with the inclusion of knowledge management in projects delivery plans, such failures can be greatly reduced or completely avoided. Software development is knowledge intensive, with different teams involved, critical business knowledge can end up in the wrong hands; the research indicates that a mature knowledge management strategy would mitigate the risk of compromising the organization's intellectual property.

The success of a project is determined through delivering the right product; on time; at the right price. This research indicates that through prudent knowledge management, projects are able to meet business demands. The analysis also shows that knowledge management ensures the safeguarding of project artefacts such as business process documentations, system configurations and source code, which would otherwise be in danger of being lost. This research has shown that knowledge is a key component of the Software Development Lifecycle and that software delivery processes are about utilizing knowledge to create new knowledge or transforming knowledge from one form into other forms. With the implementation of a sound knowledge management strategy within their SDLC processes, organizations are able to secure valuable knowledge, share knowledge efficiently amongst the teams to improve turnaround times and to leverage existing knowledge to drive innovation. The significance of knowledge management in the SDLC is highlighted by the fact that without an effective knowledge management strategy the project is likely to fail. The failure is due mainly to lack of coordination; rework and inability to leverage of existing knowledge.

6.1.2 Utilizing Knowledge Management Systems to minimize project risks

With the criticality of knowledge confirmed and the importance of knowledge management proven, the research examined the value brought by using knowledge management systems within an outsourced software development environment. Knowledge management systems

support the core knowledge management processes namely: - Discovery, Capture, Sharing and Application of knowledge. The various tools and strategies were discussed which support the management of knowledge artefacts within the SDLC. The Knowledge Management Systems use IT as an enabler to support the knowledge management strategy. Systems capture knowledge making it instantly available to a wider or restricted audience depending on the rules defined, decisions that affect the progress of the outsource project are made and reviewed at the fraction of the time using real time data combined with lessons from previous projects. Knowledge Management Systems facilitate the management of knowledge and support the organization in making the best of its knowledge. The research analyzed different knowledge management systems and their application within the outsourced SDLC processes. It has been shown that the knowledge management strategy is implemented through Knowledge Management Systems and that without these systems; the implementation of the strategy would be slow, labour intensive and therefore constraining to the overall progress and success of the project. Knowledge Management Systems such as Communities of Practice establish positive team dynamics which last well after the project is completed and thus help to increase maturity and preserve the currency of the systems. Knowledge management brings repeatability to the project delivery process. Without repeatable processes projects are likely to repeat old mistakes and as such will fail.

6.1.3 Using knowledge to maintain project control across borders

With most of the IT outsource work going offshore to India and China as highlighted in the research; it is inevitable that the teams building the outsourced solution will be dispersed across widespread geographic locations. The research went on to pose the question, if knowledge is key to the delivery of outsourced software development, how is it affected by outsourcing across geographic locations and therefore how does this affect project delivery? The research examined the forces at play when teams are dispersed across geographic locations, these included proximity advantages, cultural differences, language barrier, time differences and knowledge flows. The research showed that regardless of the model employed (insource, outsource), distance brings with it challenges such as highlighted above which can undermine project delivery. Effective and efficient flows of knowledge between the teams were found to be a critical success factor in countering distance related issues. In terms of Nonaka's SECI model,

distance would hamper the exchange of knowledge, necessitating a technological solution to be implemented to bridge the gap and mitigate the issues. The research also highlighted the risk associated with the free flow of knowledge into third parties.

6.1.4 Securing organizational intellectual property

In analyzing the outsourcing model, the following risks were found to have the potential to compromise an organization's knowledge assets:-

- Loss of control of key operations.

  Outsourcing involves handing over part of an organization's operations to a third party, the organization therefore loses the capability it had to perform the outsourced functions and must depend on the outsource partner for delivery. As indicated in the study, outsourcing is moving up the value chain and critical operations are being outsourced, this implies that the inability of the outsource partner to deliver, for whatever reason, could threaten the survival of the outsourcing business. The study has confirmed loss of knowledge due to outsourcing as a major risk and that mitigating this risk would encompass managing the knowledge involved in performing the outsourced functions.

- Potential loss of intellectual property.

  The research has shown that with outsourced functions and the know-how contained in the business processes and IT applications moving outside of the control of an organization, there is no guarantee that such know-how which constitute an organization's intellectual property will not land up in wrong hands and as such compromise the organization's strategy. The study has confirmed that securing these knowledge assets is key to the survival of the organization.

- Loss of know-how.

  As indicated above, outsourcing creates dependency as there is no longer a need for an organization to keep outsourced know-how internally. The study has shown that loss of such know-how compromises the organization and puts it in a vulnerable position during contract negotiations as the service provider enjoys an upper hand and can therefore dictate the terms of the contract.

- Threat to information security and confidentiality.

  The analysis has shown that while the maturity of an outsource partner's processes can be verified, that of its partners cannot be; leading to confidential information not being handled in an appropriate manner, this could expose an organization to law suits and reputational damage. This potential risk highlights the need for a rigorous knowledge management process that incorporates all service providers in the value chain.

- Quality problems and Hidden costs.

  The need to outsource highlighted in the research pointed to a variety of drivers including access to rare skills and the need to cut costs and gain capacity, however in pursuit of cost cutting an organization can lose its ability to deliver good quality as its service provider battles to understand the organization's processes because of poor transition management (knowledge management). This research study has shown that the cost of poorly managed transition can lead to higher costs resulting in the outsource transaction costing a lot more than anticipated.

The research advanced a number of strategies that organizations could consider to mitigate the above risks including:-

- Using insourcing as a strategy to mitigate loss of control and countering information security and confidentiality risks. Insourcing implies the use of internal capability or a subsidiary which complies with the culture, governance and ethos of the organization.

- Ensuring that the contract is based on a Win-Win philosophy.

  The research has shown that when the service provider shares equally in the benefits of the outsourcing transaction and is not carrying a strain, he is unlikely to indulge in activities that could harm the relationship and compromise organizational knowledge.

- Investing in a creating a culture of knowledge sharing.

  The study discussed a model for sharing knowledge and listed a number of barriers to knowledge sharing. A strategy to counter these barriers would result in the free-flow of information and knowledge reducing the risk of dependency through single points of failure.

- Establishing a formal governance process.

  The research highlighted a conceptual framework that could be utilised to guide the processes and decisions around outsourcing. The framework ensures that all inputs and circumstances are taken into consideration in deciding on an appropriate outsourcing model namely: - insource, outsource, onshore, offshore.

The ISC model was presented as a framework that could be used to secure knowledge assets across the extended enterprise, the different domains were proposed as a way to ensure that the security of the knowledge assets is guaranteed across the span of the organization including participating service providers.

6.1.5 Making outsourcing succeed as a business strategy

Research into the evolution of outsourcing revealed that outsourcing as a concept has been growing constantly fuelled mainly by globalization and the fall in ICT infrastructure costs. The indication of the success of the outsourcing concept was supported by the evidence that, while organizations were initially only outsourcing back-office operations, the trend is rapidly moving towards the outsourcing of high value operations. Predictions from analysts are that the growth of the IT outsourcing industry will continue for some time to come and that new business models are emerging to take advantage of the opportunities presented by outsourcing. It was indicated that while offshore outsourcing started as a cost cutting measure it has evolved into a strategic sourcing option being utilised by organizations of varying sizes. The outsourcing models presented showing the different permutations of offshore/onshore, outsource/insource, indicate a maturing industry that is able to cater for the different business requirements. Research presented on the business drivers of outsourcing indicate that outsourcing is used by organizations to satisfy a strategic need, drivers such as cost reduction; enabling business agility and focusing on core business support confirm the important role played by outsourcing in enabling business strategy

The evidence advanced by the research shows the significant role played by knowledge management in the success of offshore outsourced software projects. The research also answers the research question posed and there proves the hypothesis.

6.2 Research limitations

Literature review was chosen as an approach underpinning this research study. While every attempt has been made to ensure that the findings of this study represent a balanced and an objective view, it is however acknowledged that the literature review approach used does have limitations. Following are some of the research limitations inherent with the literature review design:-

- **Representativity of the sample**

  As indicated before (paragraph 2.4.3); insight gained in a literature review is highly dependent on the extent to which the sample represents the bigger population. By their very nature, Fortune 500 type conglomerates such as Boeing, Chevron and Google are likely to be the subject of research regarding their outsourcing activities. The success or failure of strategies used by this relatively small sample of organizations influence perceptions and therefore literature on outsourcing. When measuring success of software development projects, research organizations such as those referenced in this study are likely to sample mainly the same type of organizations thus creating a skewed picture.

- **Extent to which the subject being researched is documented**

  Knowledge Management, Outsourcing and Software development are topics that have normally been researched independently. While there is an abundance of research material covering these topics individually, the body of knowledge that combines them in the manner this study is attempting to do is fairly limited. Limitation on available literature can narrow the scope of the findings rendering it inapplicable to general use.

- **Selective interpretation**

  The outsourcing discourse is characterized by a sizable amount of research material that is either sponsored or produced directly by parties that are driven more by commercial interest than scholarly output. The result of this commercialization of knowledge is research that is geared towards selling rather than informing. A study based on this research is likely to be influenced by the direction that the reference material is pushing.

- **The need for validation**

  Based on textual evidence presented and theoretical insight, this research makes a strong case for knowledge management to be incorporated into the strategies of outsourcing organization. There is however a need to validated these findings through empirical means.

On a general note; knowledge management and software development are mature disciplines, however outsourcing on the other hand is constantly evolving and therefore research that combines these topics can be likened to shooting at a moving target. Because the subject of outsourcing has different angles to it which include social, geographic, economic and political considerations, the dominant view at a time tends to be influenced by current events, an example being if a war breaks out between India and Pakistan, the view would be that outsourcing is risky, if there is an economic crisis leading to a recession in the western world, the view would be that outsourcing takes away jobs from the locals, if there is a rise in Indian salaries, the view would be that outsourcing will be uneconomical and therefore not viable.

The public discourse on outsourcing is also clouded by a lot of material that is written by vendors and associated parties with an agenda to drive economic interest rather than advance knowledge on the subject, the challenge is therefore to sift through this to ensure that only valid views prevail.

6.3 Recommendations

While the research was not exhaustive, there are however a few important recommendations that can be made with regard to knowledge management in a software development outsourcing relationship. Following are some of the key considerations that organizations looking to outsource their software development offshore need to observe:-

- Knowledge management needs to be planned upfront and factored in as part of the outsource delivery strategy. The Discovery, Capture, Sharing and Utilization of knowledge cannot be left to the project manager to figure out or to the technical staff to resolve, it needs to be part of the key deliverable of a successful outsource engagement.

- Knowledge is the deliverable. Organizations need to put in place ways of measuring whether knowledge gains are taking place in the outsource relationship; benefits from such knowledge gains would include quality improvements in the organization's processes, the complexity of their product set and increase in market share.
- Offshore outsourcing has an element of risk in it and one of the key risks is the loss of intellectual capital, organizations need to ensure that the hype of vendor presentations does not blind them to these risks and that strategies are put in place to mitigate these upfront.
- Offshoring of software development is a viable option that can be successfully leveraged; however organizations need to equip themselves with the right knowledge before undertaking such an initiative.

6.4 Possibilities for future research

Knowledge has played a critical role in uplifting developing countries such as India and Brazil to their current state as critical players in the knowledge economy, it is the researcher's belief that with the focus on building its ICT infrastructure and knowledge potential, South Africa could leverage on the language and time zone advantages to become a formidable player in the knowledge space and an even bigger trading partner of the developed world. This research has dealt with only a part of the huge potential to leverage knowledge for economic gain, more needs to be done in understanding the trends of the global economy, how knowledge influences these and how organizations and countries can leverage knowledge for the benefit of their economies.

Commodities such as gold are gradually losing their shine on the world economy stage; knowledge is largely replacing these as the goods of exchange. For South Africa to compete and succeed in the future, she needs to rapidly start shifting its economic dependency away from commodities into knowledge enabled services. Taking into consideration the fact that the employer of the future will not be the big conglomerates that are seen today but rather the small business and the informal sector, a point of critical interest and a burning research topic is how can South Africa through the partnership and collaboration with the state, academia, the private sector and social formations build a sustainable model that could transform the small business sector into a hub of knowledge based products and services. This would be a big task but one that is achievable if tackled in phases with each phase building on the achievements of the

previous one. Of importance would be the extent to which fast increasing desktop computer processor speeds, constant reduction in desktop computer prices, availability of affordable bandwidth and open source software could be leverage to support this model.

Bibliography

Accenture, 2008, *Outsourcing Market Research – China and the World*, [Online] Available: http://www.accenture.com/NR/rdonlyres/9950CC5D-F76F-4986-8520-3176C9D473A6/0/161ChinaOutsourcingReport_FinalDec08.pdf

Alavi M, Leidner DE, 2001, *Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues*, MIS Quarterly, Vol. 25(1) pp107

Application Executive Council, 2007, *Boosting the Productivity of Requirements Capture*, [Online] Available: www.aec.executiveboard.com

Bahli B, Rivard S, 2005, *Validating Measures of Information Technology Outsourcing Risk Factors,* Omega: The International Journal of Management Sciences, Vol. 33 pp175

Basili V, Rombach HD, 1991, *Support for Comprehensive Reuse*, IEE Software Engineering Journal, Vol. 22(4)

Becerra-Fernandez I, Gonzalez A, Sabherwal R, 2004, *Knowledge Management – Challenges, Solutions and Technologies*, Prentice Hall, New Jersey

Blackler F, 1995, *Knowledge, Knowledge work and Organizations: An Overview and Interpretation,* Organization Studies Vol. 16

Boisot M, 1999, *Knowledge Assets – Securing Competitive Advantage in the Information Economy*, Oxford University Press, New York.

Booth WC, Colomb GG, Williams JM, 2008, *The Craft of Research 3ed*, University of Chicago Press, Chicago

Bravard J, Morgan R, 2006, *Smarter Outsourcing – An Executive guide to Understanding, Planning and Exploiting successful outsourcing relationships*, Pearson Education, Great Britain

Brenner W, Zarnekow R, Wiitig H, 1998, *Intelligent Software Agents – Foundation and Application*, Berlin, Springer

Carmel E, 1999, *Global Software Teams: Collaborating Across Borders and Time Zones*, New Jersey, Prentice Hall.

Carmel E, Agarwal R, 2002, *The maturation of offshore sourcing of the information technology work*, MIS Quarterly Executive, Vol. 1(2), p65-78.

Castells M, 2000, *The Rise of the Network Society – The Information Age: Economy, Society and Culture Volume 1 2ed*. Massachusetts, Blackwell.

Chakrabarty S, 2006, *The Journey to the New Lands: Utilizing the Global IT Workforce through Offshore-Insourcing* in Young P, Huff S (Eds), *Managing IT Professionals in the Internet Age,* Vol. 1 p277-318, IGI Publishing, PA.

Chen MS, Han J, Yu PS, 1996, Data Mining: *An Overview from a Database Perspective*, IEEE Transactions on Knowledge and Data Engineering, Vol. 8 (6) pp866

CIO Magazine, 2001, *Knowledge Management: Collaborating for a competitive edge*, www.cio.com

Cockburn A, 2002, *Agile Software Development*, Addison Wesley, Boston.

Cohen L, Young A, 2006, *MultiSourcing – Moving Beyond Outsourcing to Achieve Growth and Agility*, Harvard Business School Press, MA.

Davenport TH, Prusak L, 1998, *Working Knowledge*, Harvard Business School Press, MA.

Davis-Blake A, Hui P, 2003, *Contracting Talent for Knowledge-Based Competition* in Jackson S, DeNisi A, Hitt M (Eds), *Managing Knowledge for Sustained Competitive Advantage – Designing Strategies for Effective Human Resource Management*. Jossey-Bass, CA.

Dignum V, 2002, *A Knowledge Sharing Model for Peer Collaboration in the Non-Life Insurance Domain,* [Online] Available: http://igitur-archive.library.uu.nl/math/2008-1015-201522 /dignum_02_knowledge.pdf

Eltschinger C, 2007, *Source Code China – The New Global Hub of IT Outsourcing*, John Wiley, New Jersey

Evans I, 2004, *Achieving Software Quality through Team Work*, Artech House, Boston

Florida RL, 2002, *The Rise of the Creative Class and how it's transforming work, leisure, community and everyday life*, Basic Books, New York

Friedman TL, 2005, *The World is Flat – A brief history of the twenty-first century*, Farrar, Straus and Giroux, New York

Gartner, 2003, Offshore Insourcing vs Offshore Outsourcing, ID Number: COM-20-4614, www.gartner.com

Gartner, 2008, *Gartner on Outsourcing, 2008-2009, ID Number: G00164206*. www.gartner.com

Gartner, 2009, *Vendor Risk Management is becoming core Enterprise Competency*, [Online] Available: http://www.gartner.com/it/page.jsp?id=1015714

Gibson JL, Ivancevich JM, Donnelly JH, 2003, *Organizations - Behavior, Structure, Processes*, McGraw-Hill, New York

Gilley MK, Rasheed A, 2000, *Making More by Doing Less: An Analysis of Outsourcing and its Effects on Firm Performance*, Journal of Management, Vol. 26, p763

Gold T, 2005, *Outsourcing software development offshore – Making it work*, CRC Press, Florida

Gottschalk P, 2005, *Strategic Knowledge Management Technology*, Idea Group, PA

Gottschalk P, Solli-Saether H, 2007, *Managing IT Outsourcing Performance*, IGI Global, New York

Gray P, Tehrani S, 2003, *Technologies for Disseminating Knowledge* in Holsapple CW (Eds), *Handbook on Knowledge Management Vol. 2*, Springer, New Delhi

Gregory PH, 2010, *CISA - Certified Information Systems Auditor Exam Guide*, McGraw-Hill, New York

Grover V, Davenport TH, 2001, *General Perspective on Knowledge Management: Fostering a Research Agenda*, Journal of Management Information Systems Vol. 18(1) pp5.

Haldin-Herrgard T, 2000, *Difficulties in the Diffusion of Tacit Knowledge in Organizations*, Journal of Intellectual Capital Vol. 1(4), pp357-369

Hansen MT, Nohria N, Tierney T, 1999, *What's Your Strategy for Managing Knowledge?,* Harvard Business Review Vol. 77(3), pp106

Harris S, 2008, *All In One CISSP Exam Guide 4th Edition*. McGraw-Hill, New York

Henninger S, 1997, *Case-Based Knowledge Management Tools for Software Development*, Automated Software Engineering Vol. 4

Housel T, Bell AH, 2001, *Measuring and Managing Knowledge*, McGraw Hill, New York

Huysman M, De Wit D, 2003, *A Critical Evaluation of Knowledge Management Practices* in Ackerman M, Pipek V, Wulf, V (Eds), *Sharing Expertise – Beyond Knowledge Management*, MIT Press, Massachusetts

ISACA, 2010, *The Information Systems Audit and Control Association*, www.isaca.org

Jones C, 1996, *Patterns of Software Systems Failure and Success*. Thompson Computer Press, Massachusetts

Jonker J, Pennink B, 2010, *The Essence of Research Methodology – A Concise guide for Master and Phd students in Management Science*, Springer, Berlin

Karamouziz F, Young A, 2004, *India maintains its offshore leadership position, Gartner Research Note - ID Number: G00121630*, http://www.gartner.com

Kazi AS, 2005, *Knowledge Management in the Construction Industry: A Socio Technical Perspective*, Idea Group, London.

Kehal HS, Singh VP, 2006, *Outsourcing and Offshoring in the 21st century – A socio-economic perspective*, Idea Group, London

King R, 2008, *The new economics of outsourcing*, [Online] Available: http://www.businessweek.com/technology/content/apr2008/tc2008043_531737_page_2.htm

KPMG, Swamy & Associates, 2008, *Knowledge Processing Outsourcing*, [Online] Available http://us.kpmg.com/Rutus_Prod/Documents/12/KPOwhitepaper_USsize.pdf

Kulanga GK, McCaffer R, 2001, *Measuring Knowledge Management for Construction Organizations*, Engineering, Construction and Architectural Management, Vol. 8 pp346.

Laplante PA, 2007, *What Every Engineer Should Know About Software Engineering*, CRC Press, Florida

Laudon KC, Laudon JP, 2006, *Managing the Digital Firm*, Pearsons Education, New Jersey

Lave J, Wenger E, 1991, *Situated Learning: Legitimate Peripheral Participation*, Cambridge University Press, MA

Lewis M, 2006, *IT Application Service Offshoring – An Insider's Guide*, Sage, New Delhi

Maier R, 2004, *Knowledge Management Systems – Information and Communication Technologies for Knowledge Management 2ed.*, Springer, Berlin

Martin P, Tate K, 2001, *Getting Started in Project Management*, John Wiley & Sons, New York

McDermott RM, 1999, *Why Information Technology inspires but cannot deliver Knowledge Management*, California Management Review, Vol. 41(4), pp103

McDermott RM, 2000, *Community Development as a natural step*, Knowledge Management Review, November/December

Meyer B, Joseph M, Gotel O, 2007, *Software Engineering Approaches for Offshore and Outsourced Development*, Springer, Berlin

Mitra S, 2008, *The Coming death Of Indian Outsourcing*, [Online] Available: http://www.forbes.com/2008/02/29/mitra-india-outsourcing-tech-enter-cx_sm_0229outsource.html

Mouton J, 2001, *How to succeed in your Master's & Doctoral Studies – A South African Guide and Resource Book*, Van Schaik, Pretoria

Munch J, Heidrich J, 2005, *Tool-Based Software Project Control* in Chang SK (Ed), *Handbook of Software Engineering and Knowledge Engineering Vol.3 – Recent Advances*, World Scientific, Singapore

Nonaka I, 1991 , *The Knowledge Creating Company*, Harvard Business Review, Vol. 69, pp96

Nonaka I, Takeuchi H, 1995, *The Knowledge Creating Company - How Japanese Companies Create the Dynamics of Innovation*, Oxford University Press, New York

O'Dell C, Elliot S, 2003, *Achieving Knowledge Outcomes* in Hosapple CW (Eds), *Handbook on Knowledge Management Vol. 2*, Springer, New Delhi

Pettigrew A, 1972, *Information Control as a Power Resource*, Sociology Vol. 6, pp187.

PMP Research, 2004, *Outsourcing is in*, [Online] Available: www.pmpresearch.com/downloads/pmpresearch_WisingUp.pdf

Pressman RS, 2001, *Software Engineering – A Practitioner's approach 5ed*, McGraw Hill, New York

Prusak L, 1996, *The Knowledge Advantage,* Strategy and Leadership, Vol. 24(2)

Ribiere VM, 2005, *Building a Knowledge-Centered Culture: a Matter of Trust* in Stanskosky M, (Eds), *Creating The Discipline of Knowledge Management*, Elsevier, Oxford

Ruggles R, 1998, *The State Of The Nation: Knowledge Management In Practice*, California Management Review, Vol. 40(3), p80-89.

Sahay S, Nicholson B, Krishna S, 2003, *Global IT Outsourcing - Micro-studies across borders,* Cambridge University Press, Cambridge

Scholl R, Chohan S, 2003, *Offshore insourcing vs. Offshore outsourcing*, Gartner research note (ID Number: COM-20-4614), http://www.gartner.com

Senge PM, *The Fifth Discipline – The Art & Practice of The Learning Organization*. Doubleday/Currency, New York

Sibbet D, 1997, *75 Years of Management Practice: 1922 – 1997*, Harvard Business Review Special Supplement, September – October.

Software Engineering Institute (SEI), 2006, *CMMi for Development version 1.2*, Carnegie Melon University Press, Pittsburgh

Sommerville I, 2007, *Software Engineering 8Ed.*, Pearson Education Press, Essex

St. Amant K, 2010, *Understanding IT Outsourcing- A Perspective for Managers and Decision Makers*, Business Science Reference, Vol. 1

Standish Group, 2009, *CHAOS summary 2009*, [Online] Available http://www.standishgroup.com/newsroom/chaos_2009.php

Stewart TA, 1999, *Intellectual Capital – The New Wealth of Organizations*, Currency/DoubleDay, New York

Turner JR, 2007, *Gower Handbook of Project Management 4ed*, Gower Publishing, Hampshire

Vagadia B, 2007, *Outsourcing to India – A Legal Handbook*, Springer, Berlin

Waterson P, Clegg C, Axtell C, 1997, *The Dynamics of work organization, Knowledge and Technology during Software Development*, International Journal of Human-Computer Studies Vol. 46, p79

Watson RT, 1999, *Data Management – Databases and Organizations 2ed*, Wiley Press, New York

Wei C, Piramuthu S, 2003, *Knowledge Discovery and Data Mining* in Hosapple CW (Eds), *Handbook on Knowledge Management Vol. 2*, Springer, New Delhi

Weick K, 1995, *Sensemaking in Organizations*, Sage Publications, London

Wenger EC, McDermott R, Snyder WM, 2002, *Cultivating Communities of Practice*, Harvard Business School Press, MA

Wiig K, 1999, *The Knowledge Management Handbook*, CRC Press, Florida

Wilcocks L, Lacity MC, 1998, *Strategic Sourcing of Information Systems - Perspectives and Practices,* Wiley, UK

Zack MH, 1999, *Developing a Knowledge Strategy*, California Management Review, Vol. 41(3), pp125